Chapter 2
Game Analytics – The Basics

Anders Drachen, Magy Seif El-Nasr, and Alessandro Canossa

Take Away Points:

- Overview of important key terms in game analytics.
- Introduction to game telemetry as a source of business intelligence.
- In-depth description and discussion of user-derived telemetry and metrics.
- Introduction to feature selection in game analytics.
- Introduction to the knowledge discovery process in game analytics.
- References to essential further reading.

A. Drachen, Ph.D. (✉)
PLAIT Lab, Northeastern University, Boston, MA, USA

Department of Communication and Psychology, Aalborg University, Aalborg, Denmark

Game Analytics, Copenhagen, Denmark
e-mail: andersdrachen@gmail.com

M. Seif El-Nasr, Ph.D.
PLAIT Lab, College of Computer and Information Science,
College of Arts, Media and Design, Northeastern University,
Boston, MA, USA
e-mail: magy@neu.edu; m.seifel-nasr@neu.edu

A. Canossa, Ph.D.
College of Arts, Media and Design, Northeastern University,
Boston, MA, USA

Center for Computer Games Research, IT University of Copenhagen,
Copenhagen, Denmark
e-mail: a.canossa@neu.edu

2.1 Analytics – A New Industry Paradigm

Developing a profitable game in today's market is a challenging endeavor. Thousands of commercial titles are published yearly, across a number of hardware platforms and distribution channels, all competing for players' time and attention, and the game industry is decidedly competitive. In order to effectively develop games, a variety of tools and techniques from e.g. business practices, project management to user testing have been developed in the game industry, or adopted and adapted from other IT sectors. One of these methods is *analytics*, which in recent years has decidedly impacted on the game industry and game research environment.

Analytics is the process of discovering and communicating patterns in data, towards solving problems in business or conversely predictions for supporting enterprise decision management, driving action and/or improving performance. The methodological foundations for analytics are statistics, data mining, mathematics, programming and operations research, as well as data visualization in order to communicate insights learned to the relevant stakeholders. Analytics is not just the querying and reporting of BI (Business Intelligence) data, but rests on actual analysis, e.g. statistical analysis, predictive modeling, optimization, forecasting, etc. (Davenport and Harris 2007).

Analytics typically relies on computational modeling. There are several branches or domains of analytics, e.g. marketing analytics, risk analytics, web analytics – and game analytics. Importantly, analytics is not the same thing as data analysis. Analytics is an umbrella term, covering the entire methodology of finding and communicating patterns in data, whereas analysis is used for individual applied instances, e.g. running a particular analysis on a dataset (Han et al. 2011; Davenport and Harris 2007; Jansen 2009).

Analytics forms an important subset of, and source of, **Business Intelligence** (BI) across all levels of a company or organization, irrespective of its size. BI is a broad concept, but basically the goal of BI is to turn raw data into useful information. BI refers to any method (usually computer-based) for identifying, registering, extracting and analyzing business data, whether for strategic or operational purposes (Watson and Wixom 2007; Rud 2009). Common for all business intelligence is the aim to provide support for decision-making at all levels of an organization – as defined by Luhn (1958): "the ability to apprehend the interrelationships of presented facts in such a way as to guide action towards a desired goal." In essence, the goal of BI – and by extension game analytics – is to provide a means for a company to become data-driven in its strategies and practices.

In the context of the ICT industry, BI covers a variety of data sources from the *market* (benchmark reports, white papers, market reports), the *company* in question (QA reports, production updates, budgets and business plans) and not the least the *users* (players, customers) of the company's games (user test reports, user research, customer support analysis). These sources of BI operate across temporal (historical as well as predictive) and geographical distances as well as across products. **Game analytics** is a specific application domain of analytics, describing it as applied in the

context of game development and game research. The direct benefit gained from adopting game analytics is support for decision-making at all levels and all areas of an organization – from design to art, programming to marketing, management to user research. Game analytics is directed at both the analysis of the *game as a product*, e.g. whether it provides a good user experience (Law et al. 2007; Nacke and Drachen 2011) and the *game as a project*, e.g. the process of developing the game, including comparison with other games (benchmarking).

Just like "regular" analytics in the IT sector in general, game analytics is concerned with all forms of data that pertains to game business or research – not just data about user behavior or from user testing. This is a common misconception because the analysis of user behavior has been an important driver for the evolution of game analytics in the past decade, and because in the cousin fields: web analytics and mobile analytics – two of the strongest sources of inspiration for game analytics – customer behavior analysis is a key area. Game analytics is a young domain, where there has yet to emerge a standard set of key terms and processes. Such standards exist in other sub-domains of analytics, e.g. web analytics, providing models for establishing such frameworks in game analytics in the future (WAA 2007).

To sum up, game analytics is business analytics adapted to the specific context of games. This by extension makes the domain of game analytics fairly broad and too cumbersome a topic to be treated in detail in any one book. Indeed, business intelligence, analytics, big data, data-driven business practices and related topics are the subject of numerous books, white papers, reports and research articles, and it is not possible in this chapter – nor this book – to provide a foundation for the entire field of game analytics. In this chapter a brief introduction is provided focusing on the topics that the chapters in this book focus on: while this book covers a range of topics on game analytics, the chapters are generally – but not exclusively – focused on two aspects of game analytics:

1. **Telemetry:** The chapters in this book focus on a particular source of data used in game analytics: *telemetry*. Telemetry is data obtained over a distance, and is typically digital, but in principle any transmitted signal is telemetry. In the case of digital games, a common scenario sees an installed game client transmitting data about user-game interaction to a collection server, where the data is transformed and stored in an accessible format, supporting rapid analysis and reporting.
2. **Users:** Data on user behavior is arguably one of the most important sources of intelligence in game analytics, and user-oriented analytics is one of the key application areas of game analytics. Users in this context have a dual identity, as players of games and as customers. However, game analytics also covers areas such as production and technical performance, but these are less comprehensively covered in this book (but see for example Chaps. 6 and 7).

One of the main current application area of game analytics is to *inform Game User Research* (GUR), which the chapters in this book also reflect. GUR is the application of various techniques and methodologies from e.g. experimental Psychology, Computational Intelligence, Machine Learning and Human-Computer Interaction to evaluate how people play games, and the quality of the interaction

between player and game. This is a big topic in game development in its own right
(see e.g. Medlock et al. 2002; Pagulayan et al. 2003; Isbister and Schaffer 2008;
Kim et al. 2008). The practice of GUR follows many of the same tenets as user-
product testing in other ICT sectors, but with a general focus on the user experience
which is paramount in game design (Pagulayan et al. 2003; Laramee 2005).
Essentially, GUR is a form of game analytics because the latter covers all aspects of
working with data in games contexts; but, game analytics is more than GUR. Where
GUR is focused on data obtained from users, game analytics consider all forms of
business intelligence data in game development and research.

   This chapter is intended to lay the foundation for the book and provide a very
basic introduction to game analytics. It is focused on describing the basic terminology
of the domain with a specific emphasis on user behavior analytics. The chapter is
structured in sections, as follows:

- **Section 2.2** lays out key terms and concepts in game analytics
- **Section 2.3** discusses the fundamental considerations guiding the selection of
  which user behaviors to track, log and analyze
- **Section 2.4** outlines the basics for collection and application of game telemetry
  data and the knowledge discovery process in game analytics.

   Throughout the chapter, references are provided to other chapters in the book
where topics introduced here are treated in more depth.

   On a final note, this chapter does not go into direct detail on the *benefits* of apply-
ing game analytics to game development and research. This topic is the focus of
Chap. 3, which details the benefits to all the main groups of stakeholders involved,
e.g. designer and user research. Game analytics: key terminology.

   There are many different kinds of data that can form the input streams in game
analytics, and thus game BI. However, as mentioned above, this book is generally,
but not exclusively (e.g. Chaps. 21 and 22), focused on **telemetry**.

2.1.1   Telemetry

The collection and application of telemetry has a history dating back to the nine-
teenth century where the first data-transmission circuits were developed, but today
the term covers any technology that permits measurement over a distance (derived
from Greek: tele = remote; metron = measure). Common examples include radio
wave transmission from a remote sensor or transmission and reception of informa-
tion via an IP network. **Game telemetry** is the term we use to denote any source of
data obtained over distance, which pertain to game development or game research.
There are many popular applications of telemetry in games, including remote moni-
toring and analysis of game servers, mobile devices, user behavior and production.
The source of telemetry most strongly represented in this book is user telemetry, i.e.
data on the behavior of users (players), for example on their interaction with games,
purchasing behavior, physical movement, or their interaction with other users or

applications (Thompson 2007; Drachen and Canossa 2011; Mellon 2009; Bohannon 2010; Fields and Cotton 2011).

Game telemetry data can be thought of as the raw units of data that are derived remotely from somewhere, for example an installed client submitting data about how a user interacts with a game, transaction data from an online payment system or bug fix rates. In the case of user behavior data, code embedded in the game client transmits data to a collection server; or the data is collected from game servers (as used in e.g. online multi-player games like *Fragile Alliance* (Square Enix, 2007), *Quake* (id Software, 1996+) and *Battlefield* (EA, 2002)) (Derosa 2007; Kim et al. 2008; Canossa and Drachen 2009).

The actual data being transmitted follow different naming conventions depending on the field of research or application domain that people are applying the data to. This can cause some confusion when reading research articles on game analytics. The essence is that telemetry is measures of the *attribute* of *objects* (or *items*). Objects in this case should be understood broadly – an object can be virtual objects, people, processes, etc. – anything that has one or more measureable attributes. For example, the location of a player character as it navigates a 3D environment. In this case the location is the attribute, the player character the object. Conversely, the length of customer service calls generated from a newly released patch in an MMORPG sees the length of the calls as the attribute of the customer service calls.

In order to work with telemetry data, the attribute data needs to be *operationalized*, which means having to decide a way to express the attribute data. For example, deciding that the locational data tracked from player characters (or mobile phone users) should be organized as a number describing the sum of movement in meters. Operationalizing attribute data in this way turns them into variables or features – the term varies depending on the scientific field. In Experimental Psychology the term *variable* is usually used, and thus this is the term that is generally seen in articles and conference presentations on telemetry used in game user research. In Computer Science the term *feature* is often used, and thus this is the term used in data mining articles. This is just a general guideline – naming conventions vary considerably because game analytics is not a domain with established standards, so care must be taken when consulting the literature on game analytics (such as it is). Finally, variables/features have a specific *domain*. The domain is the set of all possible values – defining the domain is essentially what operationalizing attribute data is all about. For example, a binary domain allows only two values (e.g. 0 or 1).

2.1.2 Game Metrics

Raw telemetry data can be stored in various database formats (see Chaps. 6, 7 or 12), which are ordered in such a way that makes it possible to transform the data into various interpretable measures, such as average completion time as a function of individual game levels, average weekly bug fix rate, revenue per day, number of

daily active users, and so forth (see Chaps. 4 and 12). These are called **game metrics**. Game metrics are, in essence, interpretable measures of something. They present the same potential advantages as other sources of BI, i.e. support for decision-making in companies. Metrics can be variables/features and vice versa, or more complex aggregates or calculated values, for example the sum of multiple variables/features. To take an example: telemetry data from a shooter like *Quake* could include data on the location of the player avatar in the virtual environment, the weapons used, and information on whether every shot hits or misses, etc. These are different attributes, and they can be converted into variables/features such as "number of hits" or "number of misses" with a domain from 0 to 1,000 (with 1,000 being the biggest number of hits scored for a specific level). In turn, these simple variables/features can form the basis for analysis, e.g. calculating the hit/miss ratio for each level or map in Quake (e.g. "hit/miss ratio is 1.2 on average for the "Albatross" map"). An alternative is to use the variables/features "playerID", "session length" and "points scored" to calculate the metric "points scored per minute" for each player. These kinds of measures, which are based on calculations involving several variables/features, are usually referred to as "game metrics". However, there is no standard terminology widely accepted in game analytics, so be prepared for variations.

Additionally, it is important to note that most types of analysis and analytics software do not separate between a simple variable/feature or metric, or a more complex metric – when it comes to inputting measures into an analysis, they will follow the same naming standard as specified by the software. For example, in the statistics package SPSS (or PASW in newer generations) all measures of an object or objects are called "variables". It does not matter whether this variable is a simple operationalization or a number calculated using a dozen such variables.

Metrics are usually calculated as a function of something. The typical unit is time, but can also be game build (version), country, progression in a game, or number of players or players' ID, to name a few. All metrics are bound to some sort of timeframe, and this will always be from a past period – we cannot (yet) collect telemetry from the future. Telemetry based on past performance is generally referred to a "rear-view data", and form the basis of traditional BI. However, it is possible to run predictive analyses based on historical data, which can generate metrics for future behavior, e.g. expected sales figures, expected churn rate, expected number of players, expected behavior of specific user groups, etc. However, these will always be based on predictions with a specific uncertainty attached, whereas collected telemetry data – if collected correctly – are facts.

To sum up, and provide a tentative and sufficiently broad definition, a *game metric is a quantitative measure of one or more attributes of one or more objects that operate in the context of games.* Translated into plain language, this definition clarifies that a game metric is a quantitative measure of something related to games. For example, a measure of how many daily active users a social online game has; a measure of how many units a game has sold last week; a measure of the number of employee complaints the past year; task completion rates in a production team for a specific title, etc. – are all game metrics, because they relate directly to some aspect of one or more games.

Conversely, metrics that are unrelated to the games context, for example the revenue of a game development company last year, the number of employee complaints last month, etc., are *business metrics*. The distinction can be blurry in practice, but is essential to separate what is purely business metrics with those metrics that relate to games, of which a number are unique to game development (in how many other IT sectors can "number of orcs killed per player" be a business metric?).

While the term game metrics has become something of a buzzword in game development in recent years, metrics have arguably been around for as long as digital games have been made, but the application of game telemetry and game metrics to drive data-driven design and development has expanded and matured rapidly in the past few years across the industry.

2.1.3   Non-Telemetry-Based Metrics

The term game metrics is often used as a synonym for measures based on operationalized game telemetry data, but it is worth noting that a game metric does not need to be derived from telemetry data. The connection between telemetry and game metrics is commonly made in game development due to the inspiration of the use of the term "metric" in web analytics and mobile analytics, which have been among the primary inspirational sources for game analytics.

A game metric is a quantitative measure of something related to games, but this does not specify that a particular method (i.e. telemetry) has to be used to obtain the measure. For example, the "average completion time" for a specific game level during a ten-person user test can be measured using a stopwatch or obtained via telemetry software. This does not change the fact that both resulting measures are metrics (but using a stopwatch introduces a potential problem with measurement accuracy). In this book, the term game metric is generally used for telemetry-derived measures, but as detailed in e.g. Chaps. 21 and 22, metrics can be derived from other sources of data.

2.1.4   Game Metrics: Types and Classes

Mellon (2009) categorized game metrics into three types, based on an expansion and slight redefinition of which the following categories of game metrics can be defined:

1. **User metrics:** (labeled "player metrics" in Mellon 2009) These are metrics related to the people, or users, who play games, from the dual perspective of them being either *customers*, i.e. sources of revenue or *players*, who behave in a particular way when interacting with games. The first perspective is used when calculating metrics related to revenue, e.g. average revenue per user (ARPU), daily active users (DAU) or when performing analyses related to revenue, e.g. churn analysis, customer support performance analysis or micro-transaction

analysis (see Chaps. 4 and 12). The second perspective is used for investigating how people interact with the actual game system and the components of it and with other players, i.e. focusing on in-game behavior. Examples of metrics are: total playtime per player, average number of in-game friends per player or average damage dealt per player; and common analyses include time-spent analysis, trajectory analysis, or social networks analysis (Chaps. 17, 18 and 19). The data used to generate player metrics typically originate in telemetry, notably from game clients, game servers or online payment processing tools (Chaps. 6 and 7).

The vast majority of the published knowledge about game analytics is based on player metrics, and this book is also biased towards the application of player metrics for game development. This focus on player metrics is driven at least in part by the increased focus on Game User Research (GUR) (see below and Chaps. 16, 21, 22, 25, 26 and 27 or 31 and 32 for a specific view on metrics and learning games) and the increasing popularity of social online games (Chap. 4).

2. **Performance metrics:** These are metrics related to the performance of the technical and software-based infrastructure behind a game, notably relevant for online or persistent games. Common performance metrics include the frame rate at which a game executes on a client hardware platform, or in the case of a game server, its stability. Performance metrics are also used when monitoring changing features or the impact of patches and updates on how well the client executes. A simple performance metrics known since the first game was programmed is the number of bugs found – per hour, day, week or any other timeframe. Performance metrics are heavily used in QA to monitor the health of a game build. It is also one of the most mature areas of game analytics, because the methods employed are derived from traditional software performance and QA techniques and strategies. See Chaps. 6, 7 and 23 for more on performance metrics.

3. **Process metrics:** These are metrics related to the actual process of developing games. Game development is to a smaller or greater degree a creative process, which – similar to other creative areas in IT – has necessitated the use of agile development methods. In turn, this has prompted the development of ways of monitoring and measuring the development process. For example, by combining task size estimation with burn down charts, or measuring the average turnaround time of new content being delivered, type and effect of blocks to the development pipeline, and so forth. Similar to performance metrics, a number of process metrics and the associated management and monitoring methods are adopted and/or adapted from the methods and strategies in use outside the games sector. See Chaps. 6, 7 and 23 for more on process metrics.

2.1.5   A Closer Look at User Metrics

"You are no longer an individual, you are a data cluster bound to a vast global network" – trailer for the game "Watch Dogs"(Ubisoft) presented at E3 in 2012

The above quote is pretty spot on when it comes to how game analytics view users in games – they are clusters of data about the attributes of a particular object (the player), and its connection to the larger "network" of the game. User metrics is a common source of business intelligence in a range of sectors, and this is also the case for game development and research. The vast majority of knowledge published in the past 5 years on game analytics is based on user metrics, and especially user behavior telemetry. This is not surprising given that the users (players) are alpha and omega for the success of a games title – games are products that are focused on delivering user experience, and being able to analyze how users interact with games is a prime source of information about the degree of success of a games' design to deliver engaging experiences (Medlock et al. 2002; Kim et al. 2008; Nacke and Drachen 2011). User metrics therefore deserve a closer inspection.

A key feature of games – whether digital or not – is that they are **state machines**. What this means is that during play, a person creates a continual loop of actions and responses which keep the game state changing (Salen and Zimmerman 2003). The game engages the user and often loops the player through the same steps over and over again, keeping the user engaged over a period of time. This period of time arguably varies, but compared to e.g. purchasing a product from an online store, a game session takes longer time and generates a lot more actions from the user and reactions from the system – i.e. more state changes. This means that they generate more user-behavior data than most software applications, with terabytes of data easily being accumulated in a brief period of time (Drachen and Canossa 2011; Weber et al. 2011). This goes for both perspectives of the user: customer and player.

User metrics derived from games have been classified by their applicability across games by considering three levels of applicability: **generic metrics**, which apply across all digital games (total playtime per player, number of started game sessions); **genre specific metrics**, which are applicable to a specific genre, e.g. Role-Playing Games (RPGs) (character progression, number of quests/missions completed), and **game specific metrics**, which are specific to individual games, i.e. unique features e.g. the average number of white tarantulas killed in *Tomb Raider: Underworld (Eidos Interactive, 2008)*, average number of times players chose each of the three endings in *Mass Effect 3 (Electronic Arts, 2012)*. This system of classification is useful for research purposes, but a more development-oriented classification system, which serve to funnel user metrics in the direction of three different classes of stakeholders, is suggested here (shown in Fig. 2.1).

- **Customer metrics:** Covers all aspects of the user as a customer, e.g. cost of customer acquisition and retention. These types of metrics are notably interesting to professionals working with marketing and management of games and game development.
- **Community metrics:** Covers the movements of the user community at all levels of resolution, e.g. forum activity. These types of metrics are useful to e.g. community managers.
- **Gameplay metrics:** Any variable related to the actual behavior of the user as a player – inside the game, e.g. object interaction, object trade, and navigation in
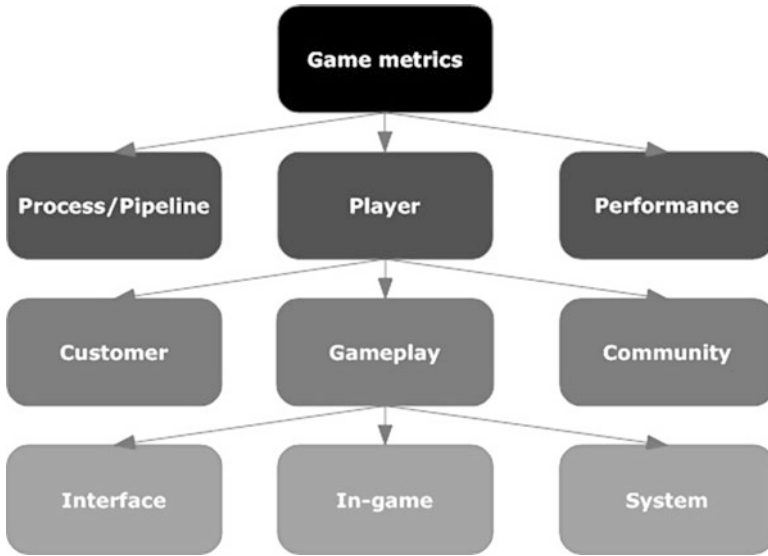
Fig. 2.1 Hierarchical diagram of game metrics emphasizing user metrics

the environment. Gameplay metrics are the most important to evaluate game design and user experience, but are furthest from the traditional perspective of the revenue chain in game development, and hence are generally under prioritized. These metrics are useful to professionals working with design, user research, quality assurance, or any other position where the actual behavior of the users is of interest.

2.1.5.1 Customer Metrics

As a **customer**, users can download and install a game, purchase any number of virtual items from in-game or out-of-game stores and shops, spending real or virtual currency, over shorter or longer timespans. At the same time, customers interact with customer service, submit bug reports, requests for help, complain, or otherwise interact with the company. Users can also interact with forums, whether official or not, or any other kind of social interaction platform, from which information about the users, their play behavior and how satisfied they are with the game, can be mined and analyzed (see Chap. 7). Customers also have properties. They live in specific countries, generally have IP-addresses, and sometimes we details about them such as their age, gender and email address. Combining this kind of demographic information with behavioral data can provide powerful insights into a games´ customer base. Chapter 4 describes a number of examples of customer metrics.

2.1.5.2   Community Metrics

Players interact with each other. This interaction can be related to gameplay – e.g. combat or collaboration through game mechanics – or social – e.g. in-game chat. Player-player interaction can occur in-game or out-of-game, or some combination thereof. For example, sending messages bragging about a new piece of equipment using a post-to-Facebook function. In-game, interaction can occur via chat functions, out-of-game via live conversation (e.g. using Skype) or via game forums.

These kinds of interactions between players form an important source of information, applicable in an array of contexts. To take an example, social networks analysis of the user community in a free-to-play (F2P) game can reveal players with strong social networks, i.e. players who are likely to retain a big number of other players in the game via creating a good social environment. A good example is guild leaders in MMORPGs. Mining chat logs and forum posts can provide information about problems in a game's design. For example, data mining datasets derived from chat logs in an online game can reveal bugs or other problems (see Chap. 7 for an example). Monitoring and analyzing player-player interaction is important in all situations where there are multiple players, but especially in games that attempt to create and support a persistent player community, and which have adopted an online business model, e.g. many social online games and F2P games. These examples are just the tip of a very deep iceberg, and the collection, analysis and reporting on game metrics derived from player-player interaction is a topic that could easily take up a book on its own. See Chaps. 4, 7 and 21 for more on this topic.

2.1.5.3   Gameplay Metrics

This sub-category of the user metrics is perhaps the most widely logged and utilized type of game telemetry currently in use in the industry. Gameplay metrics are measures of player behavior, e.g. navigation, item- and ability use, jumping, trading, running and whatever else players actually do inside the virtual environment of a game (whether 2D or 3D). Five types of information can be logged whenever a player does something – or is exposed to something – in a game: *What is happening? Where is it happening? At what time is it happening?* In addition, when multiple objects (e.g. players) interact: *to whom is it happening?*

Gameplay metrics are particularly useful to game user research for informing game design. They provide the opportunity to address key questions, including whether any game world areas are over- or underused, if players utilize game features as intended, or whether there are any barriers hindering player progression. This kind of game metrics can be recorded during all phases of game development, as well as following launch (Isbister and Schaffer 2008; Kim et al. 2008; Lameman et al. 2010; Drachen and Canossa 2011).

As a player, users can generate thousands of behavioral measures over the course of a just a single game session – every time a player inputs something to the game system, it has to react and respond. Accurate measures of player activity can include dozens of

actions being measured per second. Consider, for example, player in a typical fantasy MMORPG like *World of Warcraft* (Blizzard, 2003): measuring user behavior could involve logging the position of the player's character, its current health, mana, stamina, the time of any buffs affecting it, the active action (e.g. running, swinging an axe), the mode (in combat, trading, traveling, etc.), the attitude of any MOBs towards the player, the player character name, race, level, equipment, currency etc. – all these bits of information flowing from the installed game client to the collection servers.

From a practical perspective (e.g. for naming different groups of metrics in a way that makes them easily searchable), it can be useful to further subdivide gameplay metrics into the following three categories:

- **In-game:** Covers all in-game actions and behaviors of players, including navigation, economic behavior as well as interaction with game assets such as objects and entities. This category will in most cases form the bulk of collected user telemetry.
- **Interface:** Includes all interactions the user (player) performs with the game interface and menus. This includes setting game variables, such as mouse sensitivity, monitor brightness.
- **System:** System metrics cover the actions game engines and their sub-systems (AI system, automated events, MOB/NPC actions, etc.) initiate to respond to player actions. For example, a MOB attacking a player character if it moves within aggro range, or progressing the player to the next level upon satisfaction of a pre-defined set of conditions.

To sum up, the sheer array of potential measures from the users of a game (or game service) is staggering, and generally analysts working in game development try to locate the most essential pieces of information to log and analyze. This selection process imposes a bias but is often necessary to avoid data overload and to ensure a functional workflow in analytics (for more on this topic see Chaps. 3, 4, 6, 7, 9, 12 and 14).

2.1.6 Example Gameplay Metrics Across Game Types

Up to this point the discussion about user attributes has been at a fairly abstract level, because it is nigh-on impossible to develop classes of which user metrics it makes sense to develop in which types of games. This not just because games do not fall within neat design classes (games share a vast design space but do not cluster at specific areas of it), but also because the rate of innovation in design is high, which would rapidly render recommendations invalid. In this section some examples of useful gameplay metrics are provided for different game genres. Despite being nebulous, genre definitions are commonly used to provide e.g. readers of game reviews some idea about which type of game we are dealing with. For example, labeling *Skyrim (Bethesda Softworks, 2011), Deus Ex Human Revolution (Eidos Interactive, 2000)* and *Diablo III (Blizzard Entertainment, 2012)* as Role-Playing Games, due to

the ability of the user to modify the character played during the game, irrespective of the many other differences in their gameplay and style. Genres make for useful terms when defining what "school" of mechanics drive a game.

In essence, it is the mechanics (and thus by inference genre, but keeping in mind that genres are nebulous), and the underlying business model (e.g. traditional one-shot vs. F2P) which determines what types of player telemetry that can be logged and analyzed.

2.1.6.1   Action Games

Action games are generally focused on quick reflexes, accuracy, timing etc., over to more explorative-heavy games. Usually single character/avatar played. Examples: Pinball games, racing, FPS' and TPS'.

Useful gameplay metrics: In general anything that relates to the reflex-based mechanics.

First-Person Shooters (FPS)

First-Person Shooters are shooter games, i.e. focused on combat involving projectile weapons of some kind, with the camera looking out of the eyes of the player. Fast paced games, reflex-based play, can include strategic elements, heavily reliant on engagement. Examples: *Unreal* (GT Interactive, 1998), *Quake* (GT Interactive, 1996), *Halo* (Microsoft Studios, 2001). Note how team-based FPS' like *Team Fortress 2* (Valve, 2010) track a wealth of player behaviors and provide them back to the players.

Useful gameplay metrics: Weapon use, trajectory, item/asset use, character/kit choice, level/map choice, loss/win [quota], heatmaps, team scores, map lethality, map balance, vehicle use metrics, strategic point captures/losses, jumps, crouches, special moves, object activation. AI-enemy damage inflicted + trajectory. Possibly even projectile tracking.

Third-Person Shooters (TPS)

Third-Person Shooters are shooter games, i.e. focused on combat involving projectile weapons of some kind, with the camera from a third-person perspective relative to the player avatar. Includes shoot'em up-games, arcade-style games where the player controls a central avatar who kills massive numbers of enemies. Fast paced games, reflex-based play, can include strategic elements. Examples: *Project X* (Team17 Software, 1992), *Starfighter* (Micros, 1984), *Aerial Command* (Croft Soft Software, 1994).

Useful gameplay metrics: as for FPS + camera angle, character orientation.

Racing

Racing games are games where the player controls a vehicle. Usually reliant on reflex gameplay and some strategic thinking.

Useful gameplay metrics: Track choice, vehicle choice, vehicle performance, win/loss ratio per track and vehicle, completion times, completion ratio per track and player, upgrades [if possible], color scheme [if possible], hits, avg. speed different types of tracks/track shapes.

2.1.6.2 Adventure Games

Maybe two different genres – Adventure and Action-adventure – but exceptionally hard to separate. Incredibly varied – usually single-player, focused on exploration and puzzle-solving, but can also include combat, although normally not reliant on reflex-based play. Often heavy story element. Includes interactive stories. Puzzle heavy. Examples: *Deus Ex: Human Revolution* (Square Enix, 2011), *Tomb Raider: Underworld* (Square Enix, 2008). Pattern analysis is highly useful (see Chap. 12).

Useful gameplay metrics: story progression [e.g. node based], NPC interaction, trajectory, puzzle completion, character progression, character item use, world item use, AI-enemy performance, damage taken and received + source (player, mob).

2.1.6.3 Arcade

Simple mechanics, fast-paced play, generally game is never completed. Example: *Pac-Man* (Atari, 1981), *Asteroids* (Atari, 1981).

Useful gameplay metrics: trajectory, powerup usage, special ability usage, session length, stages completed, points reached, unlocks, opponent type damage dealt/received, player damage dealt/received [as applicable].

2.1.6.4 Beat'em Up

Fighting game, generally restricted to one player controlling one avatar in combat with another, but can be multi-player beyond two people. Generally players control a "humanoid" avatar. Examples: *Double Dragon* (Activision, 1988), *Tekken* (Namco Bandai, 1995).

Useful gameplay metrics: Character selection, ability use, combo use, damage dealt, damage received (per ability, character etc.), weapon usage, arena choice, win/loss ratio as a feature of character, player skill profiles.

2.1.6.5 Family Games

A game designed to be played by both adults and children together. Example: *Mario Cart* (Nintendo, 2011), *Buzz!* (Sony, 2005) Includes partygames.

Useful gameplay metrics: varies substantially – subgame selection, character/avatar selections, game mode used, in-game selections, asset use, number of players, etc. form some of the possibilities.

2.1.6.6 Fitness Games

Also called exergames. A game designed to improve people´s fitness. Often played in combination with various hardware accessories. Examples: *Yourself Fitness* (Respondesign, 2008), *Wii Fit* (Nintendo, 2007), *Dance Dance Revolution* (Kontami, 2001).

Useful gameplay metrics: session length, calories burned, exercises chosen, match between exercises shown and player actions, player accuracy in performing exercises, total playtime over X days, player hardware/exercise equipment [usually registered], player demographics [usually entered during profile creation], music tracks selected, backgrounds selected, avatar selection, powerups/content unlocked [common feature], total duration of play per user.

2.1.6.7 Music Games

Also called audiogames. A game where the players sing or where the gameplay is otherwise heavily reliant on music-related mechanics. Commonly challenge the player to follow sequences of movement or develop specific rhythms. Examples: *Singstar* (Sony, 2004).

Useful gameplay metrics: Points scored, song/track chosen, match with rhythm/auditory mechanics, difficulty setting, track vs. difficulty, track vs. errors, track vs. choices.

2.1.6.8 Platformer Games

A game focused on navigation in 2D or 3D space along platforms. Examples: *Mario* (Nintendo, 1983-), *Sonic* (Sega, 1992-), *Giana Sisters* (a.k.a. The Great Giana Sisters, Rainbow Arts, 1987).

Useful gameplay metrics: jumping, progression speed, items collected, powerups/abilities used, AI-enemy performance, damage taken + sources of damage.

2.1.6.9   RPGs

Role Playing Games are extremely varied – can be any other genre but includes crucially the ability for the player to develop the avatar/character/-s being controlled. Examples: *Diablo* (Blizzard Entertainment, 1996), *Dragon Age: Origins* (EA, 2009), *Mass Effect* (BioWare, 2007), *Eye of the Beholder* (Strategic Simulations, 1991-). Temporal and spatial analysis can be useful, see Chap. 19 for more on analysis of RPGs.

Useful gameplay metrics: character progression, quest completions, quest time to complete, asset use (resources), character ability/item use [including context of use], combat statistics, AI-enemy performance, story progression [including choices], NPC interactions [e.g. communication], ability/item performance, damage taken + sources of damage, cutscene viewed/skipped, items collected [including spatial info].

2.1.6.10   Simulation

Simulation: A very diverse category of games, where the main focus is on simulating some aspect of life or fiction, from constructing and managing cities in *SimCity* (Maxis, 1989-) to simulating life in *Spore* (EA, 2008) and *Evo* (Enix Corporation, 1992), or vehicles from cars to air planes, including combat simulators.

Useful gameplay metrics: Very hard to predict due to the sheer variety in simulation games. Asset use would be important, but depends on the specifics of the game.

2.1.6.11   Sports Games

Any game where the main focus is on the execution of sports activities. Examples: *FIFA World Manager* (Ubisoft Entertainment, 1998), *Madden NFL* (EA, 1993-), *Wii sports* (Nintendo, 2006).

Useful gameplay metrics: match types, win/loss ratios, team selection, color schemes, country chosen, management decisions [if game includes management aspects], in-match events [e.g. goal scored, fouls, tackles, length of hit], item use [e.g. club type], heatmap [density of player time spent on sections of the field], team setup/ strategy, player [in-game] selection, player commands to team/team members.

2.1.6.12   Strategy Games

Can be broadly divided into either real-time or turn-based strategy games (e.g. *Starcraft* vs. *Civilization V*) (Blizzard, 1998; 2K Games, 2010). The gameplay is focused on strategic planning and plan execution, and often the player controls multiple avatars, e.g. units. More specialized strategy games include smaller groups of units to control and a TPS/FPS view. Includes the small category of "god games"

and "puzzle games" as well as tower defense games. Metrics choices vary generally depending on whether the game is a real-time strategy game (RTS) or turn-based (TBS). Spatial analysis can be useful for these types of games see Chap. 19 for an example.

Useful gameplay metrics: all features related to player strategy and control. Generally two types of things players can build: building and units. Selections and order of selection are crucial metrics. Commands given to units, upgrades purchased, trajectory, win/loss ratio, team/race/color selection, maps used, map settings, match/game settings (usually strategy games have some settings that affect the core mechanics). Race/aspect/team chosen, time spent on building tasks vs. unit tasks.

2.1.7 Tracking Strategies

The transmission of a piece of information via a telemetry system – irrespective of whether this is in the context of user, process or performance measures – in games can occur in three fundamental ways:

1. **Event:** A pre-specified event occurs, for example, a user starts a game, a designer submits a bug fix request, a unit of a game is sold, a player fires a weapon, buys an item, etc. – any action initiated by a person or system forms an event. Event-based telemetry is based on tracking such actions and transmitting this information to a collection server.
2. **Frequency:** Rather than being triggered by the occurrence of a specific event, information can be recorded following a specific frequency. For example, when tracking the trajectory of player avatars through virtual environments, we can record the location of the avatar once per second, as a compromise between precision and bandwidth constraints. Frequency-based recording of telemetry is generally used when the attribute of the object being tracked is always present, e.g., a player character in an MMORPG always has a position in the world when playing.
3. **Initiated:** Sometimes the game analyst wants to enable and disable the tracking of a specific attribute, rather than having a telemetry system autonomously submitting tracked information based on some pre-defined command. For example, it may not be necessary to record player avatar trajectories all the time, but only when updates or patches are pushed to the users. Having the ability to turn on and off recording of specific attributes can be useful in these situations.

There are different strategies available for the recording, transmission and storage of game telemetry. For example, sampling can be employed to reduce the overall amount of data being collected and thereby to reduce costs of running analyses. This topic is described in Chap. 9. Similarly, there are different options for how to physically handle the client-side recording and transmission of telemetry to collection servers, a topic discussed in Chaps. 6, 7 and 12.

2.2   Ethics and Privacy

A key issue when working with user telemetry is the question of individual privacy and ethics. Current technology allows for the collection of detailed information on users from digital games, and combined with information harvested via collaboration with third parties, highly detailed patterns of behavior can be mined, e.g. information about the habits and preferences of individuals. Collected behavioral telemetry can be used to generate player profiles, and correlated with personal data forms a means of targeting players with marketing messages that are highly specific. The existence of such datasets is controversial given their confidential nature and the potential illegal access and use. Because data are valuable, they are also traded, and this can lead to user information migrating to the hands of people who will employ the knowledge unethically.

Currently the typical practice in the game industry is to keep the user (or consumer) data confidential and not sell or share this data. Furthermore, most analysis is run on anonymized data, so the identities of the users are not shown to the analyst working with the data, although basic information such as demographics might be known. It is however not the norm that users are clearly informed that their behavior is being tracked, and there is rarely a chance to opt out of tracking and still play the game in question. The digital analytics association has developed the Web Analyst Code of Ethics, which are directly applicable to game analytics (http://www.digitalanalyticsassociation.org/?page=codeofethics), however, there is currently no widely agreed – upon standard in game analytics, and ethics therefore remain a largely grey and undefined area in the field. Some of these issues have been discussed in the interview with the independent developer Nifflas, Chap. 20.

2.3   Selecting User Behaviors

Having covered the basic categories of game metrics the next question that arises is: given the array of possible variables/features to track from a digital game, which of these should we track? There is no one answer to this question. Like all other applications of BI, game analytics is a highly context-dependent process, perhaps especially so for computer games, because of the substantial variation in design, business models, target audience, revenue drivers, value chains etc. However, as noted above, games that share design features, e.g. free-to-play social online games for *Facebook*, will likely share metrics related to these shared features that are useful across these games – but not outside of them.

In comparison, *process* and *performance* metrics are more generalizable across games and companies, because there is a substantial overlap in the methods employed in game development across the industry, e.g. a common use of agile frameworks, similar marketing strategies, and so forth (see any book on game development and –management for more on game production) (e.g. Laramee 2005). When selecting user metrics, especially for titles with complex gameplay and thus

hundreds of possible user actions and -interactions, the question becomes more difficult to answer. In this section we outline some of the fundamental consider-ations in *feature selection* (selecting what user behaviors to track and analyze), more in-depth discussion can be found in Chap. 13. For feature selection and key metrics in social online games/F2P specifically, see Chaps. 4 and 35 for an inter-view with Junebud or Fields and Cotton (2011).

2.3.1 Balancing Cost and Benefit

User-oriented game analytics can have a variety of purposes, but can broadly be divided into:

- *Strategic analytics*, which target the global view on how a game should evolve based on analysis of user behavior and the business model.
- *Tactical analytics*, which aim to inform game design at the short-term, for example an A/B test of a new game feature.
- *Operational analytics*, which target analysis and evaluation of the immediate, current situation in the game. For example, informing what changes should be made to a persistent game to match user behavior in real-time.

Operational and tactical analytics to an extent deal with technical and infrastruc-ture issues, whereas strategic analytics is more focused on merging user telemetry data with other user data and/or market research.

The first thing to be aware of when deciding on a strategy for how to approach user telemetry is the existence of these three types of user-oriented game analytics, and the kinds of input data they require. The second aspect to consider is the dimin-ishing returns on the logging of behavioral telemetry.

In a situation with infinite resources, it is possible to track, store and analyze every single user-initiated action – every fraction of a move of an avatar, every button press, all purchases made, every single chat message, all the server-side system information – even all keystrokes. Doing so will likely cause bandwidth issues, and will require substantial resources to add the message hooks into the game code, but in theory, this brute-force approach to game analytics is possible. However, it leads to very large datasets, which in turn leads to huge resource requirements in order to transform and analyze them (Han et al. 2011; Kim et al. 2008; Drachen et al. 2009). For example, tracking the weapon type, range, damage done, target, whether the target was killed or not, the weapon modifications chosen by the player, the position of the player and target, the trajectory of the bullet, etc. will provide the possibility for a very in-depth analysis of weapon use in an FPS. However, the key metrics to calculate in order to evaluate weapon balancing could just be range, damage done and the frequency of use of each weapon. Adding a number of additional variables/ features may not add any new relevant insights, or may even add noise or confusion to the analysis. Similarly, it may not be necessary to log behavioral telemetry from all players of a game, but only a percentage. This is of course not the case when it comes to sales records (see Chap. 9 for more on sampling).

In general, if selected correctly, the first variables/features that are tracked, collected and analyzed will provide a lot of insights into user behavior. As more and more detailed aspects of user behavior are tracked, costs of storage, processing and analysis increase but the rate of added value from the information contained in the telemetry data diminishes.

What this means is that there is a cost-benefit relationship in game telemetry, which basically describes a simplified **theory of diminishing returns** (from Economic theory): Increasing the amount of one source of data in an analysis process will yield a lower per-unit return.

A classic example in Economic literature is adding fertilizer to a field. In an unbalanced system (under-fertilized), adding fertilizer will increase the crop size, but after a certain point this increase diminishes, stops and may even reduce the crop size. Adding fertilizer to an already balanced system does not increase crop size, or may reduce it.

Fundamentally, game analytics follows a similar principle. An analysis can be optimized up to a specific point given a particular set of input features/variables, before additional (new) features are necessary. Additionally, increasing the amount of data into an analysis process may reduce the return, or in extreme cases lead to a situation of negative return due to noise and confusion added by the additional data.

There can of course be exceptions – for example, the cause of a problematic behavioral pattern, which decreases retention in a social online game, can rest in a single small design flaw, which can be hard to identify if the specific behavioral variables related to the flaw are not tracked.

2.3.2 User Experience Versus Monetization

The fundamental goal of game design is to create games that provide a good user experience. However, the fundamental goal of running a game development company is to make money. Aligning these two goals is vital, especially in the F2P game situation where there is no up-front investment from the customers.

In this situation, the underlying drivers for game analytics are twofold: (1) ensuring the user experience, in order to acquire and retain customers; (2) ensuring that the monetization cycle generates revenue.

It is possible to design F2P games that provide a good user experience but which absolutely fail in prompting the users to make any purchases. Similarly, it is possible to design a F2P which includes all the tricks in the book to make users to invest real-world money in the game. These two extremes have a hard time standing on their own, and therefore user-oriented game analytics must inform both design and monetization at the same time.

This approach is exemplified by companies who have been successful in the F2P marketplace, e.g. Zynga, Kiloo and Wooga, who use analysis methods like A/B
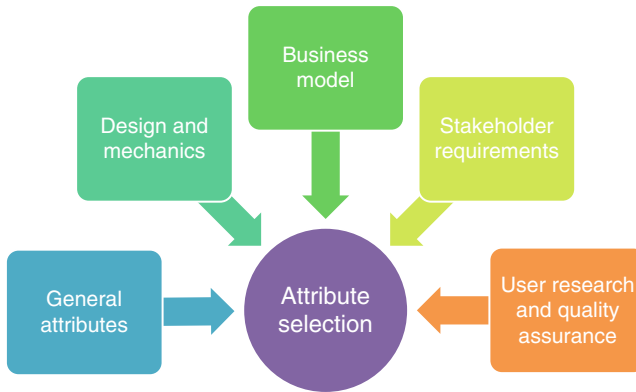
Fig. 2.2 The drivers of attribute selection for user behavior attributes. Given the broad scope of application of game analytics, a number of sources of requirements are in play

testing to evaluate whether a specific design change increases both retention and monetization (see Chaps. 4 and 5 for an interview with Zynga, Chap. 24 for an interview with Kiloo and Fields and Cotton 2011).

2.3.3 Feature Selection of User Behavior Telemetry

In real life we rarely have the resources to track and analyze all possible user behaviors, which necessitates an approach to analytics which considers cost-benefit relationships between on one side the resources required for tracking, storing and analyzing user telemetry/metrics, on the other hand the value of the insights obtained (Mellon 2009).

Following this line of reasoning, the minimum set of attributes that should be tracked, stored and analyzed about users in a computer game context is comprised of (Fig. 2.2):

1. **General attributes:** The attributes that are shared for users (as customers and players) across all games. These form the core metrics which can always be collected, for any computer game, e.g. when a user starts playing a game, stops playing, a userID, etc. For examples, see e.g. Chaps. 14, 18 and 19.
2. **Core mechanics/design attributes:** The essential attributes related to the core of the gameplay and mechanics of the game. For example attributes related to time spent playing, virtual currency spent, number of opponents killed, etc. Defining the core mechanics attributes should be based directly on the key gameplay mechanics of the game, and provide information that allows inferences to be made about the user experience. For example, whether players are progressing as planned, if flow is sustained, death ratios, level completions, point scores, etc. For examples, see e.g. Chaps. 14 and 17.

3. **Core business attributes:** The essential attributes related to the core of the business model (e.g. F2P) of the company. For example, logging every time a user purchases a virtual item, establishes a friend connection in-game, country of origin, recommends the game to a *Facebook* friend, attributes related to retention, virality and churn, etc. See Chaps. 4 and 12 for more on business-related metrics (or Fields and Cotton 2011).

In addition to these three, there can be an assortment of *stakeholder requirements* that need to be considered. For example, management or marketing may place a high value on knowing the number of Daily Active Users (DAU) (Chap. 4). Such requirements may or may not align with the categories mentioned above.

Finally, if there is any interest in using telemetry data for *user research/user testing* and *quality assurance* (e.g. recording crashes and crash causes, hardware configuration of client systems, and notably game settings), it may be necessary to augment to attributes on the list of features accordingly. Chapters 6 and 7 provides insights on this area.

When building the initial attribute set and planning the metrics that can be derived from them, it is vital that the selection process is as well-informed as possible, and includes all the involved stakeholders. This minimizes the need to go back to the code and embedding additional hooks at a later time – which is a waste that can be eliminated with careful planning. That being said, as the game evolves during production as well as following launch (whether a persistent game or through DLCs/patches), it will typically be necessary to some degree to embed new hooks in the code in order to track new attributes and thus sustain an evolving analytics practice.

Sampling is another key consideration. It may not be necessary to track every time someone fires a gun, but only 1% of these. Chapter 9 discusses sampling in detail and we will therefore not delve further on this subject here, apart from noting that sampling can be an efficient way to cut resource requirements for game analytics.

2.3.4   Pre-selecting Attributes

A final consideration is the extent to which attribute set selection can be driven by pre-planning, i.e. by defining the game metrics and analysis results (and thereby the actionable insights) we wish to obtain from user telemetry and select attributes accordingly. This is certainly possible to an extent, but the lack of an explorative component adds the risk of missing important patterns in user behavior that cannot be detected using the pre-selected attributes. This problem is exasperated in situations where the game metrics and analyses are also pre-defined – for example relying on a set of Key Performance Indicators (KPIs) – which eliminates the chance of finding any patterns in the behavioral data not detectable via the pre-defined metrics and analyses. In general, striking a balance between the two situations – tracking and analyzing everything vs. pre-selecting KPIs – is the best solution. See e.g. Chaps. 6, 7 and 14 for examples of pre-selection in practice. In Chap. 8, interview with Unity Technologies, the holy grail of dynamic feature tracking is discussed.

It is worth noting that when it comes to user behavior analytics, we are working with human behavior, which is notoriously unpredictable, at least in games contexts. This means that predicting user analytics requirements can be problematic, and forms the basis for the use of both explorative (i.e. we look at the user data to see what patterns they contain) and hypothesis-driven methods (i.e. we know what we want to measure and know the possible results, just not which one is correct), in e.g. Game User Research. These approaches are described in more detail in Chaps. 13 and 14, and by e.g. Pagulayan et al. (2003); Isbister and Schaffer (2008) and Drachen and Canossa (2011).

2.4 Telemetry Analysis and Reporting

In the above sections the fundamental terminology of game analytics has been introduced, and an overview presented of the different types of data that can form the input to the game analytics process. We now turn to the process of collecting and utilizing game telemetry. These are topics that are described in the remainder of this book, and this section will therefore briefly introduce the general steps in the game analytics process, and provide references to chapters where the different topics are treated; and the references and suggested readings in those chapters provide a guideline for further reading on the various topics. An important topic not covered here is how to integrate analytics in the business and culture of an organization. See Chaps. 6 and 7 for discussion on this topic.

The game analytics process follows the standard process for *knowledge discovery* in data (Berry and Linoff 1999; Larose 2004; Witten et al. 2011), which is widely used in data-driven analytics to discover useful knowledge from data. Knowledge discovery can be described in a number of phases or steps, which are fundamentally cyclic in nature, i.e. the result of an analysis cycle can feed into the next cycle. This is one way of continually optimizing the discovery process. The knowledge discovery process is described in Chap. 12, here we present a brief overview, with phases adapted to the context of game development and the focus on user telemetry as the data source (Fig. 2.3). The systems used to enable knowledge discovery are in Business Intelligence generally referred to as Decision Support Systems (DSS) or Knowledge Discovery Systems (KDS) depending on the specific aim.

1. **Attribute definition:** The first step in the process is defining the objectives, and the requirements, the result of the discovery process must fulfill. During this phase the user attributes to track are selected, as well as the tracking strategy (event, frequency or initiated). Domains for each attribute are defined, and *goals* for each domain defined. For example, it may be a goal that the maximum playtime for a game is set to 20 h (i.e. the game should not take any longer to complete). During this phase, strategies for balancing pre-defined metrics and results are balanced against the requirement for being able to explore and drill-down/across/through datasets (see Chap. 12 for an introduction to game data mining, Chap. 14 for an introduction to explorative telemetry work).

Fig. 2.3 The phases of the standard knowledge discovery process adapted to the context of game analytics

2. **Data acquisition:** Once the attribute set has been defined, it is implemented in the telemetry system the company uses. If no such system exists, one will have to be either purchased or a service agreement entered. There are fundamentally three ways to obtain a telemetry system: (1) develop in-house, (2) purchase a license, or (3) purchase access to a software-as-service solution. There are at the time of writing about two-dozen companies worldwide offering telemetry solutions for games. Several of these are solutions developed for e.g. business analytics or web analytics, but are also applicable to some, and in a few cases all, types of games. There are unfortunately no comprehensive guides or reviews of telemetry providers currently available, and a degree of research is therefore needed to locate the solution best suits the requirements. Chapters 6 and 7 describe two examples of telemetry systems built at Bioware and Sony Online Entertainment respectively. Chapter 10 provides an example of a flexible, open-source RestAPI for tracking user telemetry from games, aimed at small-medium sized games.
3. **Data pre-processing:** During this step, incoming telemetry data are transformed and loaded into a database structure (see Chap. 12 for a short overview of SQL

and NoSQL solutions), from where they are accessible for analysis. Additionally, data are cleaned and otherwise made ready for analysis. Chapter 12 describes data pre-processing in general, and Chap. 7 provides more specific examples.

4. **Metrics development:** following pre-processing, the attribute data are transformed into variables/features and metrics. This can be done automatically (e.g. KPIs) or manually.

5. **Analysis and evaluation:** During this step, cases and features are selected as required by the analysis in question. Sampling can also be applied to minimize resource requirements (see Chap. 9). The chosen analysis is run and a model generated of the results (see Chap. 12). Furthermore, results are evaluated and it is checked if the model reaches the required objectives.

6. **Visualization:** The results are visualized in a way that is functional given the stakeholder they aimed at, following principals of knowledge visualization (Tufte 1983, is the classic text on representation and visualization of quantitative data). Chapters 18 and 19 describe visualization of user behavior telemetry in detail. Additionally, Chap. 17 focuses on spatial metrics and visualization.

7. **Reporting:** The discovered knowledge is presented to the relevant stakeholders, e.g. a designer. The reporting/presentation should be done in such as way that the stakeholders can understand, interpret and act on the result.

8. **Knowledge deployment:** The knowledge is deployed in the organization. This will often see the initiation of a new discovery cycle.

There is a lot more to say and write about the fundamental process of knowledge discovery as it applies to game analytics, but the above covers the basic steps. Several chapters in this book go into more detail with the different steps, and steps 7 and 8 are topics that more or less all chapters touch upon, because achieving action following analysis is a fundamental goal of industrial analytics.

2.5   Conclusions and Next Steps

In this chapter the bare-bones fundamentals of game analytics have been outlined. Important key terminology has been described setting the background for game analytics as a source of business intelligence in game development and game research. The benefits of adopting analytics have been outlined, which can be summed up the following: support for decision making – at all levels and areas of an organization (or a research project). In the above we have also discussed users and user behavior in some detail, as this is a topic of core interest in game analytics.

Finally, we introduced the different challenges in feature/variable selection, and the knowledge discovery process describing the journey from raw, untreated data to actionable insights. The remaining chapters in this book will go into more detail with the topics briefly introduced here, and beyond.

On a final note, the reference list below provides an excellent starting point for further reading on game analytics.

About the Authors

Anders Drachen, Ph.D. is a veteran Data Scientist, currently operating as Lead Game Analyst for Game Analytics (www.gameanalytics.com). He is also affiliated with the PLAIT Lab at Northeastern University (USA) and Aalborg University (Denmark) as an Associate Professor, and sometimes takes on independent consulting jobs. His work in the game industry as well as in data and game science is focused on game analytics, business intelligence for games, game data mining, game user experience, industry economics, business development and game user research. His research and professional work is carried out in collaboration with companies spanning the industry, from big publishers to indies. He writes about analytics for game development on *blog.gameanalytics.com*, and about game- and data science in general on *www.andersdrachen.wordpress.com*. His writings can also be found on the pages of Game Developer Magazine and Gamasutra.com.

Magy Seif El-Nasr, Ph.D. is an Associate Professor in the Colleges of Computer and Information Sciences and Arts, Media and Design, and the Director of Game Educational Programs and Research at Northeastern University, and she also directs the Game User Experience and Design Research Lab. Dr. Seif El-Nasr earned her Ph.D. degree from Northwestern University in Computer Science. Magy's research focuses on enhancing game designs by developing tools and methods for evaluating and adapting game experiences. Her work is internationally known and cited in several game industry books, including *Programming Believable Characters for Computer Games (Game Development Series)* and *Real-time Cinematography for Games. In addition,* she has received several best paper awards for her work. *Magy worked collaboratively with Electronic Arts, Bardel Entertainment, and Pixel Ante.*

Alessandro Canossa, Ph.D. is Associate Professor in the College of Arts, Media and Design at Northeastern University, he obtained a MA in Science of Communication from the University of Turin in 1999 and in 2009 he received his PhD from The Danish Design School and the Royal Danish Academy of Fine Arts, Schools of Architecture, Design and Conservation. His doctoral research was carried out in collaboration with IO Interactive, a Square Enix game development studio, and it focused on user-centric design methods and approaches. His work has been commented on and used by companies such as Ubisoft, Electronic Arts, Microsoft, and Square Enix. Within Square Enix he maintains an ongoing collaboration with IO Interactive, Crystal Dynamics and Beautiful Games Studio.

References and Next Steps

Berry, M., & Linoff, G. (1999). *Mastering data mining: The art and science of customer relationship management*. New York: Wiley.

Bohannon, J. (2010). Game-miners grapple with massive data. *Science, 330*(6000), 30–31.

Canossa, A., & Drachen, A. (2009). *Patterns of play: Play-personas in user-centered game development. DIGRA*. London: DIGRA Publishers.

Davenport, T. H., & Harris, J. G. (2007). *Competing on analytics: The new science of winning*. Boston: Harvard Business School Press.

Derosa, P. (2007). 'Tracking player feedback to improve game design', Gamasutra. Available at: http://www.gamasutra.com/view/feature/1546/tracking_player_feedback_to_.php

Drachen, A., & Canossa, A. (2011). Evaluating motion: Spatial user behavior in virtual environments. *International Journal of Arts and Technology*, *4*(3), 294–314.

Drachen, A., Canossa, A., & Yannakakis, G. (2009). Player modeling using self-organization in tomb raider: Underworld. In *Proceedings of IEEE Computational Intelligence in Games (CIG)* (pp. 1–8). Milan: IEEE Publishers.

Fields, T., & Cotton, B. (2011). *Social game design: Monetization methods and mechanics*. Burlington: Morgan Kauffman Publishers.

Han, J., Kamber, M., & Pei, J. (2011). *Data mining: Concepts and techniques, 3rd.* San Francisco: Morgan Kaufmann Publishers.

Isbister, K., & Schaffer, N. (2008). *Game usability: Advancing the player experience*. Burlington: Morgan Kaufman Publishers.

Jansen, B. J. (2009). *Understanding user-web interactions via web analytics*. San Rafael: Morgan & Claypool Publishers.

Kim, J. H., Gunn, D. V., Schuh E., Phillips, B. C., Pagulayan, R. J., & Wixon, D. (2008). Tracking Real-Time User Experience (TRUE): A comprehensive instrumentation solution for complex systems. In *Proceedings of the Computer-Human Interaction (CHI)* (pp. 443–451), Florence, Italy.

Lameman, B. A., Seif El-Nasr, M., Drachen, A., Foster, W., Moura, D., & Aghabeigi, B. (2010) User studies – A strategy towards a successful industry-academic relationship. In *Proceedings of future play 2010* (pp. 1–9)*. Vancouver: ACM Publishers. doi:10.1145/1920778.1920798

Laramee, F. E. (2005). *Secrets of the game business*. Hingham: Charles River Media.

Larose, D. T. (2004). *Discovering knowledge in data: An introduction to data mining*. Hoboken: Wiley-Interscience.

Law, E., Vermeeren, A. P. O. S., Hassenzahl, M., & Blythe, M. (2007). Towards a UX manifesto. In Proceedings of the 21st British HCI group annual conference on HCI 2008: People and computers XXI: HCI … but not as we know it, (pp. 205–206), Florence, Italy.

Luhn, H. P. (1958). A business intelligence system. *IBM Journal, 2*(4), 314.

Medlock, M. C., Wixon, D., Terrano, M., Romero, R. L., & Fulton, B. (2002). Using the RITE method to improve products: A definition and a case study. In *Proceedings of the Usability Professionals Association,* Orlando, Florida.

Mellon, L. (2009). Applying metrics driven development to MMO costs and risks. Versant Corporation, Tech. Rep.

Nacke, L., & Drachen, A. (2011). Towards a framework of player experience research. In *Proceedings of the 2011 foundations of digital games conference*, EPEX 11. Bordeaux, France.

Pagulayan, R., Keeker, K., Wixon, D., Romero, R. L., & Fuller, T. (2003). User centered design in games. In *The human-computer interaction handbook: Fundamentals, evolving technologies, and emerging applications* (pp. 883–903). Mahwah: L. Erlbaum Associates.

Rud, O. (2009). *Business intelligence success factors: Tools for aligning your business in the global economy*. Hoboken: Wiley. ISBN 978-0-470-39240-9.

Salen, K., & Zimmerman, E. (2003). *Rules of play. Game design fundamentals*. Cambridge, MA: MIT Press.

Thompson, C. (2007). Halo 3: How Microsoft labs invented a new science of play. *Wired Magazine*, *15*(9).

Tufte, E. (1983). *The visual display of quantitative information*. Cheshire: Graphics Press.

Watson, H. J., & Wixom, B. H. (2007). The current state of business intelligence. *Computer, 40*(9), 96. doi:10.1109/MC.2007.331.

Web Analytics Association. (2007, August 16). Web analytics definitions. URL: http://www.
    webanalyticsassociation.org/resource/resmgr/PDF_standards/WebAnalyticsDefinitionsVol1.pdf
Weber BG, John M, Mateas M, Jhala A (2011) Modeling player retention in Madden NFL 11.
    Innovative Applications of Artificial Intelligence (IAAI). AAAI Press, San Francisco
Witten, I. H., Frank, E., & Hall, M. A. (2011). *Data mining: Practical machine learning tools and
    techniques*. The Morgan Kaufmann Series in Data Management Systems (3rd ed.). Morgan
    Kaufmann.