A SYSTEMS APPROACH

ADVANCED Game Design

Michael SELLERS

Advanced Game Design

The Player's Part of the Game as a System

In Chapter 3, "Foundations of Games and Game Design," we defined games and the overall game+player system. As noted there, the playful experience exists only when both the game and the player come together, each as part of a larger system (see Figure 4.1). The game creates its own internal system with structural, functional, and thematic elements—the parts, loops, and whole that define the game.

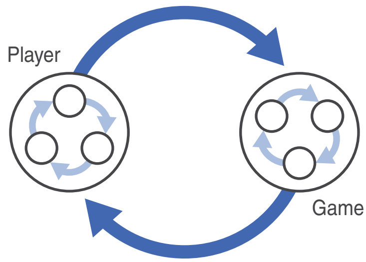

Figure 4.1 Players and games are subsystems that together create the overall game+player system. In this abstract view of the multilevel, hierarchical game+player system, each has its own structural (parts), functional (loops), and thematic (whole) elements

In this chapter we examine how the player and game create the larger game+player system via the emergent process of interactivity. Understanding interactivity enables us to go "one level down" into the internal aspects of the player subsystem—the player's mental model—and see how it is constructed as part of the experience of play.

The player's mental model corresponds to the game's internal model, discussed in Chapter 3. To build this mental model, the player and the game each act in ways that affect the other, as discussed in detail in this chapter. Through these mutual effects, the player carries out his intentions to test and affect the game's internal state. The game in turn changes its state and progressively reveals its internal parts and loops. As discussed in Chapter 2, "Defining Systems," complex systems are created with such mutual looping effects. As you will see here, this mutual looping cycle is the core concept behind interactivity, gameplay, engagement, and fun.

A Systemic Approach to Interactivity

The word *interactivity* is used throughout the games industry and in many related fields. We often treat this word as having to do with what happens when the player taps on a button or clicks on an icon in a game, but it is much deeper than that. Interactivity is central to the experience of playing a game and, more than ever before, common to the human experience. While interactivity has been cited as what differentiates games from other forms of media

(Grodal 2000), our world is far more interactive today than in the past as a result of increasing interconnections geographically and technologically. Recall from Chapter 1, "Foundations of Systems," the shift we have seen from only about 1,000 connected computing devices in 1984 to a number approaching 50 billion today, each enabling both technological and human-to-human connectivity. This is an unprecedented change in human history. And yet, despite the magnitude of this change and the ubiquity of interactivity in our lives, and despite many discussions of the topic in the fields of communications, human–computer interaction (HCI), and game design, we still lack a clear, practical definition of this core concept.

According to *Webster's*, *interactive* means "mutually or reciprocally active." This concise definition gets to the heart of what it means for something or someone to be interactive: there are two or more agents acting in relation to each other, mutually or reciprocally affecting each other by their actions. This simple definition was expanded on by Rafaeli (1988) to include the idea of two or more agents in a series of communication exchanges, where any given message is contextually related to earlier ones. This communication may be between two individuals face-to-face; in a conversation mediated by technology (for example, over the telephone); or between agents who may or may not be human, as in a human interacting with a computer game. By contrast, some authors have in the past argued that interactivity is resident only within the individual (Newhagen 2004), in the technology or medium used for communication (Sundar 2004), or even solely in human-to-human communications, on the assumption that only "the human has potential for transcending his or her programming" and that "the machine makes judgements or decisions only on the basis of its program" (Bretz 1983, 139) and is therefore somehow not truly interactive.

In game design, Chris Crawford's definition of *interactivity* reflects that of Rafaeli, cited above. Crawford characterized interactivity as "a cyclic process between two or more active agents in which each agent alternatively listens, thinks, and speaks—a conversation of sorts" (1984, 28). This turns out to be a highly useful way of looking at interactivity, and one that we will generalize here. In particular, Crawford's definition calls out the cyclic nature of any interaction, where different parts (actors) affect each other by their behavior. It begins to sound very much as if any interaction forms a system and thus can benefit from a systemic view, including parts, loops, and wholes.

Parts: Interactivity Structures

The structural parts of any interactive system are two or more actors or agents; this is true in games and any other interactive setting. In designing games, we assume that there is at least one human involved in the interactivity loop. It's possible (and often desirable) to have multiple computer-driven players in a game but only if there is also at least one human there. A game "playing itself" without human involvement can be useful for testing, but otherwise it misses out on the essential, meaningful aspect of the experience of a game as played, one that requires a human participant.

Each part in a system (here the player(s) and the game) has its own state, boundaries, and behaviors. Each part in an interactive system uses its behaviors to affect, but not wholly determine, the internal state of the others. In-game actors have an internal state such as health, wealth, inventory, speed, and so on, as well as behaviors such as talking, attacking, evading, and so on. Each agent uses its behaviors, based on its internal state, to affect others and is in turn affected by their behaviors.

Internal State

As systems in themselves, the internal states of the player and a game are necessarily complex. The human player's internal state is ultimately the totality of their current mental and emotional processing. As far as their interactions with the game are concerned, the player's internal state is their mental model of the game.[1] This includes their understanding of the following:

- Current in-game variables, such as health, wealth, country population, inventory, or whatever is relevant in the game context for their understanding
- The game state, particularly how their understanding has changed based on feedback provided from their most recent actions
- The immediate, short-term, and long-term goals within the game, including their predictions of what will happen in the game based on their actions
- The effects of past decisions and what they have learned about the game as a result of them

We will spend much of this chapter focused on these elements and the player's overall psychological state; it is, after all, for humans that we make games.

The game's internal state is the working embodiment of the game design as described in Chapter 3 and as explored in more detail throughout the rest of this book. It includes not only game-related variables and rules but also the overall event loop that determines the game's processing, when it accepts input from the player, and so on. In this chapter we treat the game's internal state as more abstract so as to focus on the interactions with the player.

Behaviors

Actions in a game are whatever the game designers enable them to be; players and non-player characters (or other actors) may talk, fly, attack, or carry out any number of other actions. These behaviors are necessarily enabled and mediated by the game, occurring as they do only within the game's context. If, for example, a game disables an ability (as by means of a "cooldown"

1. More broadly, each of us carries a mental model of every interaction we have, whether with a game, another person, or even ourselves. Here we focus on the mental models we form as part of playing a game.

timer after use), an actor cannot use that behavior until it becomes available again. Both extraordinary abilities and limitations are part of existing within the context, the magic circle, of the game environment.

Player Behaviors and Cognitive Load

A player's in-game behaviors begin as mental goals, part of the player's mental model of the game and intent within it. These behaviors must at some point become physical: the player must move a piece on the board, click on an icon, and so on. The transition from mental to physical marks the boundary from mental model to behavior.

In taking action in a game, a player typically provides input via a device, such as by tapping on a keyboard, moving or actuating (for example, clicking) a mouse or another controller, or providing a gesture (for example, tap or swipe) on a control that is sensitive to touch. In some cases, even moving their gaze to a certain part of a computer screen is a valid behavior recognized by the game.

Planning for and taking an action in a game requires intent on the player's part. Some of their limited cognitive resources have to be devoted to what they want to do, and some to the actions needed to accomplish their goal in the context of the game. The fewer cognitive resources that are needed to perform an action, the less active thought it requires, and the more natural and immediate it feels to the player.

The general term for taking up cognitive resources is *cognitive load* (Sweller 1988). The more things you are thinking about and attending to at any moment, the greater your cognitive load. Reducing how much the player needs to think about *how* to play the game reduces their cognitive load and allows them to focus instead on what they are trying to do. This ultimately increases engagement and fun. (More on this later in this chapter.)

In HCI literature, the cognitive load induced by having to think about *how* to do something is known as the combination of *articulatory* and *semantic distance* (Norman and Draper 1986). The more cognitively direct an action is—pointing with a finger being more direct than using a mouse cursor, which in turn is more direct than typing in (x,y) coordinates—the shorter the articulatory distance and the fewer cognitive resources are needed to complete it.

The semantic distance of an action is reduced when the game provides ample, timely feedback to the player and presents an easily interpreted result of an action. The more closely the feedback matches the player's understanding and intent, the fewer cognitive resources are needed to evaluate it. In a game, seeing an icon of a sword has a shorter semantic distance than seeing the letter *w* (indicating *weapon*) or the word *sword*. Seeing an animation of a building gradually being constructed is more easily evaluated than assessing a completion bar in the user interface, which is in turn more easily evaluated than seeing a text display such as "563/989 bricks placed."

The combination of these two "distances" adds to or reduces the player's cognitive load—the mental resources they must devote to understanding the game. The shorter these distances, the less the player has to actively think about the game, and the more cognitive resources they have left over to devote to the playful context within the world of the game.

Similarly in terms of game rules, the less the player has to remember about how to play the game—the fewer special cases there are in the rules—the more they can concentrate on the game itself and the shorter the semantic distance between their intent and their actions in the game. Recall the discussion of elegance in games in Chapter 2: a game such as *Go* has so few rules that the semantic distance is virtually zero, and the player is able to devote the entirety of their cognition to mentally inhabiting the game-space.

Game Behaviors and Feedback

A game's behavior must provide feedback to the player about its state. This is how the player learns how the game works and builds a mental model of it. In modern digital games, this feedback is most typically communicated via graphics (images, text, animations) and sound. This lets the player know in a timely fashion that the game's state has changed, so that they can update their mental model of the game.

While the feedback provided must be perceptible to the player—a color they cannot see or a sound they cannot hear are the same as providing no feedback at all—the game's behavior does not have to provide *complete* information about its state. This incompleteness allows for hidden state (for example, cards the game holds that the player can't see) that is a key ingredient to many game designs. Koster (2012) referred to this as the "black box" part of the game—the part that must be inferred by the player as they build a mental model of the game and that provides a great deal of the gameplay experience. Similarly, Ellenor (2014) referred to the internal game systems as "a machine that does X," meaning that the heart of the game—its internal systems—are a machine that the player discerns only through its behaviors.

It is important for you as a game designer to remember that all that the player knows about the game comes through the game's behaviors and feedback in response to the player's actions. You may want to assume that the player comes to the game with some knowledge— for example, how to operate a mouse or touch screen, how to roll dice. You must be very careful with these assumptions, though, to avoid putting the player in a position of not being able to play the game because of some missing knowledge that the game isn't going to provide.

As a player learns a game, there will be parts the player believes they know and understand well; their mental model is solid there. The more they can use this information to extend their understanding into new areas, the more easily they will learn the game. In addition, the more certain they are of some areas, the more able they will be to make predictions about areas where they are not certain or where the game is withholding some of its state information. This is where a lot of the gameplay resides, as the player tries different actions with predicted outcomes and builds their mental model based on whether those predictions were accurate.

Making Intentional Choices

It is important that an actor, whether human or computer, be able to choose its behaviors rather than fire them off at random. The choice of behavior must be based either on internal state and logic, or, in the case of a human player, on their ability to consciously choose their next action. The player must understand which actions accomplish the following goals:

- Are valid in the current context
- Have the information they need to make a choice
- Will help them accomplish their goals based on predicted outcome
- Can be decided on and selected in an appropriate amount of time

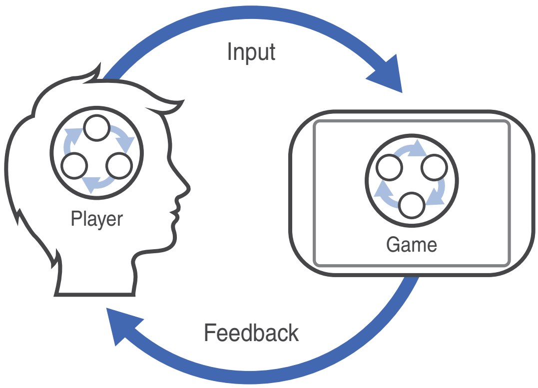

Figure 4.2 Players provide input to a game, and the games provides feedback to the player, forming a generalized interactive loop. Note that the player and the game both have internal loops as well

Interactive Game Loops

In systemic terms, the player's and the game's behaviors are the result of their internal state. Based on their state, each selects actions to take, which then affect and perturb the other's state. This drives new behavioral responses in return. The player provides *input* to the game via their behaviors, which changes the game's state. The game processes this and provides *feedback* responses that are input for the player, changing their internal state (see Figure 4.2). This creates a reciprocating loop that is the essence of interactivity. This give-and-take between the player and the game is often referred to as the game's *core loop*, a term we will define more precisely later in this chapter and revisit in Chapter 7, "Creating Game Loops."

We discussed different types of systemic loops (for example, reinforcing and balancing loops) in Chapter 2. All systems have interactive loops; the parts interact and form loops that create systems. In this case, we are focused more on interactions between the player and the game as subsystems of an overall system. We will examine these game+player interactive loops