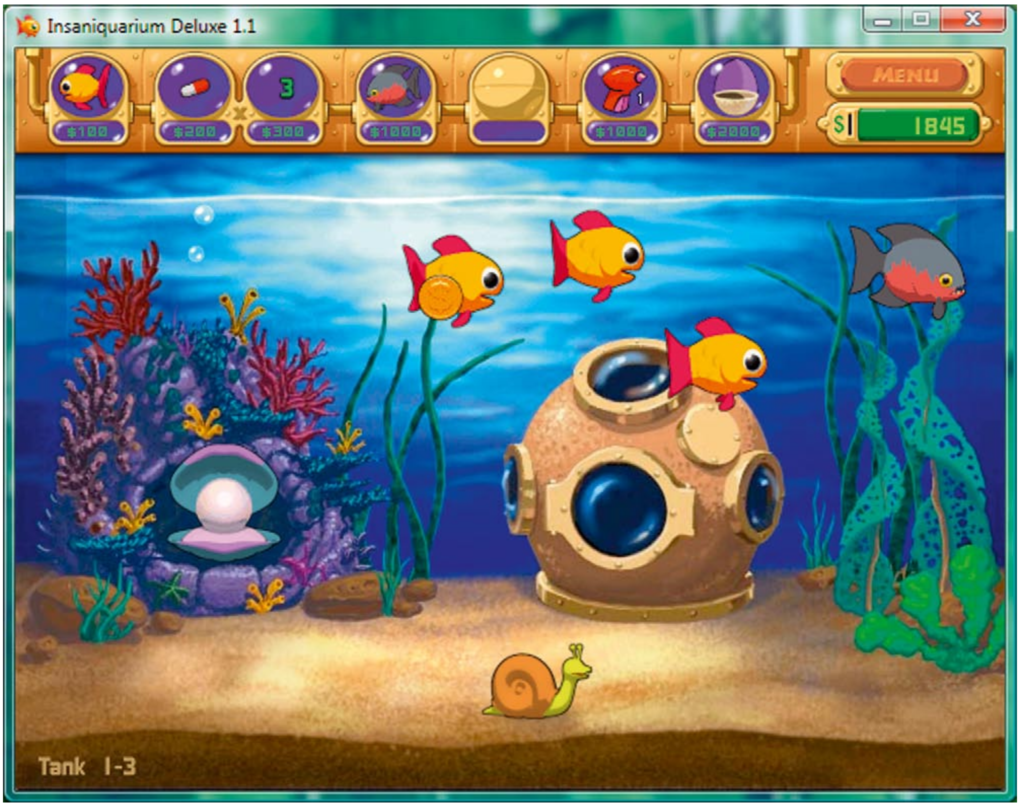
CHAPTER NINE

Chaining

Sometimes a game just needs a little something more to bring it all together. You can have a well-crafted core mechanic that works as expected and still find playing the game to be a little lifeless, a little boring. For a designer, this can be vexing. Through iteration, you've given the core mechanic the proper constraints. The interaction plays as you envisioned it. The player can move straight through the game, accomplishing small goals which lead to winning the level or the game. Yet playing the game feels as if you're painting by numbers and the game lacks a sense of tension. In cases like this, the game probably needs an additional mechanic, to add a different vector to the gameplay. By adding another vector, you offer a fork in the clear path through the game. The player can no longer simply proceed along the straight line toward the finish line without at least considering taking the other path. Done right, adding another vector adds tension and choice to the game and infuses it with more life.

There are many ways to add another vector to a game. Sometimes a game just needs more stuff—more things to look at and click on. In *Insaniquarium*, the game designers build this tension by adding more of the same. They stack similar interactions on top of one another until simply performing all of these interactions builds a requisite level of tension (or in the case of *Insaniquarium*, frenzy). You can imagine the designers, starting with the management of fish, feeding them by dropping food in front of them. That mechanic engages the player, but not fully. So the designers layer in the act of picking up coins in addition to the feeding. Now players must feed fish and pick up coins. This engages the player a bit more and adds tension in the form of more activity. Then on top of that, the designers add the alien-shooting mechanic. Performing all three interactions in concert, the game begins to achieve a sufficient level of tension. For good measure, the designers add a few more interactions: items for the player to purchase and power-ups, all with the hope of bringing the game to buzzing life. And it works. With enough interactions and stuff going on, *Insaniquarium* begins to feel dynamic, even if all of the interactions are along a similar vector (Figure 9.1). They all point the player toward the same goal. The conglomerate of these stacked interactions makes *Insaniquarium* feel a bit patchwork.

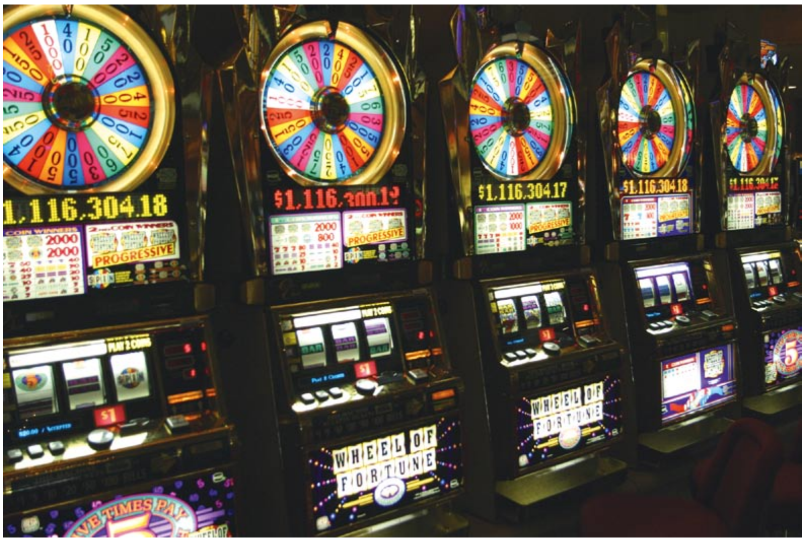
FIGURE
9.1



***Insaniquarium* brings the game to life by adding interaction upon interaction. The core of the game is feeding fish. On top of that the player collects coins, buys power-ups, shoots aliens, upgrades tools, buys new fish and clicks on pearls. Through the accumulation of clicks, the game becomes dynamic. (Reproduced by permission of PopCap Games)**

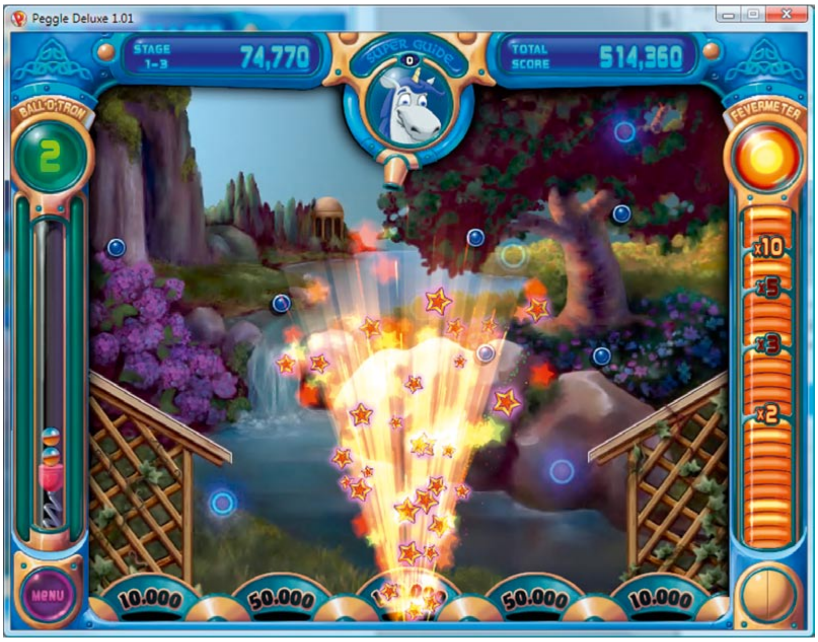
Sometimes a game comes to life with a few more points stirred into the mix. Never underestimate the importance of clear feedback and rewards for the player. In fact, some games get by almost entirely on their feedback. The pleasure in pachinko and playing slot machines derives almost entirely from the garish feedback (Figure 9.2). Sure, the random test of fate eventually grips many a gambler, but the lights and sirens draw them in. PopCap seemed to recognize this when they designed *Peggle*, their video game version of a pachinko machine. Players shoot balls at pegs and watch how they bounce around and score other pegs. Each peg awards you more points. Then at the end of the level, after you've knocked out your last peg, you are treated to a rousing rendition of Beethoven's *Ninth Symphony*, rainbows and fireworks (Figure 9.3). It's all enough to make you feel darn special and help you forget the fact that for most of the game you're simply watching a ball bounce down among a series of pegs.

FIGURE 9.2



Slot machines are studies in oversized feedback. The bells, whistles and garish lights all serve to draw the player in and make the very simple exercise of pulling a lever feel meaningful and grand. (WikiCommons[1])

FIGURE 9.3



At the end of each level in *Peggle*, the game treats the player to bursts of fireworks and streaking rainbows set to the score of Beethoven's *Ninth Symphony*. (Reproduced by permission of PopCap Games)

[1] http://commons.wikimedia.org/wiki/File:Vegas_slots.JPG, User: Mormegil

Sometimes, though, a game can't be brought to life by simply adding more clicks and stars. The game may require a different vector that actually directs player behavior and gives the experience more shape and tension. This vector needs to be at slight odds with the main gameplay. It should present a different choice or ask you, the player, to do something different from what the main game mechanic asks. Sometimes, this second vector stands in actual opposition to the main mechanic, asking you to do something which puts your success with the main mechanic at risk. This forces you to continually make the decision: do I take the easy route and do okay or take the more complex route and risk failure, but reap serious rewards?

Diner Dash: Pushing Your Luck

The chaining mechanic in *Diner Dash* provides an excellent example of a vector that runs at almost perpendicular angles to the main mechanic of ushering customers through their meal. Peter Lee, the co-founder of Gamelab and the producer of *Diner Dash*, maintains that *Diner Dash* wasn't fun until they started giving players points for every click. Despite the engaging spinning plate core mechanic, the game still felt a bit limp. But when they started giving points for each step in the process, the game came to life. As always, more feedback is a good thing. But it's not just that *Diner Dash* awards lots of points, it's how *Diner Dash* awards those points that really imbues the game with a dynamic back and forth between risk and reward.

The basic mechanic of stepping through the process of serving customers is relatively straightforward. The player learns the basics in the first level and, after a few customers, the player has the process down pat. Plus, the game won't let you bring customers the wrong order or botch the order of steps. This means the game really only gets complicated when you must serve multiple customers at the same time. When you have multiple customers, you must watch their moods and make sure you react quickly in order to move keep the line moving. But even in these cases, your best option is generally to proceed in the order the customers arrived.

This sort of straight path through play can be deadly for a game. After a bit of experimentation, players will find a game's optimal strategy and set of moves. Then they will continue to perform that set of moves over and over. They will do this even if it makes the game more boring to play. In response, frustrated game designers bemoan, "But it's more fun if you play it this way!" That very well may be, but designers can't expect players to go that extra mile and play the game the right way, just to have fun. This sort of frustration often grips first-time designers. See, the funny thing is, players don't really want to have fun. No, they want to win. Fun is a sort of happy accident by-product of playing a game in order to win.

Players want to win, and through the structure of the game, you've told them the way to win. Once players find the optimal strategy—the strategy that results in the fewest moves in proportion to the greatest score—they will continue to do that until something prevents them from doing it. Sometimes, it takes a little while to figure out the optimal strategy. But every player, from the moment they sit down with the game, is looking for the path that offers the least resistance and the greatest reward.

As a designer, ideally you want to create games that don't reduce to one optimal strategy, but instead offer an array of different strategies that offer different rewards. This can be really hard to do. Even complex systems often break down to relatively simple answers. One way to get around your game reducing to one obvious path is to offer an attractive alternative to the optimal strategy.

Diner Dash smartly avoids pitfall of having just one optimal strategy by offering a mechanic which pulls the player in an alternate direction. To add choice and tension to the gameplay, *Diner Dash* employs a chaining mechanic that subverts the straightforward march through the process of serving each customer.

Diner Dash rewards you for performing the same action multiple times in a row. Each time Flo takes an order in *Diner Dash*, you receive a base amount of 20 points. However, if you take multiple orders in a row, meaning you don't bring another customer food or clear a table, but instead just walk around taking orders, you receive a multiplier bonus that increases for each customer in the chain. So, for the first order you receive 20 points. For the second order, you receive a bonus multiplier of two, doubling your score to 40 points. For the third order in a row, you receive a bonus multiplier of three, for a total score of 60. So if you take three orders in a row, you score a total of 120 points. This is double what you would have earned if you took three orders in the normal course of play. In this way, chaining greatly increases your score. You earn a similar bonus by chaining other actions in the serving process. The longer the chain, the bigger the reward. Through these bonus points, the game heavily incentivizes you to chain.

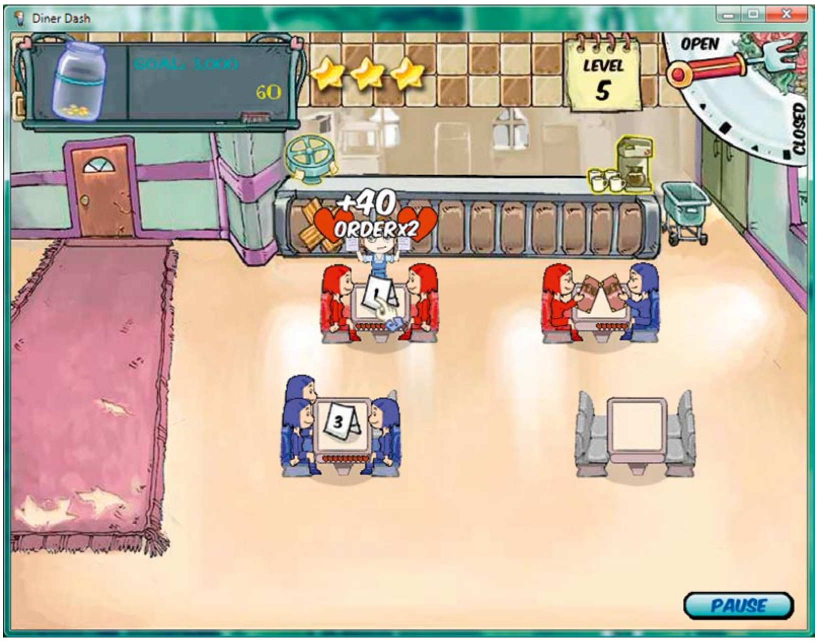


FIGURE
9.4

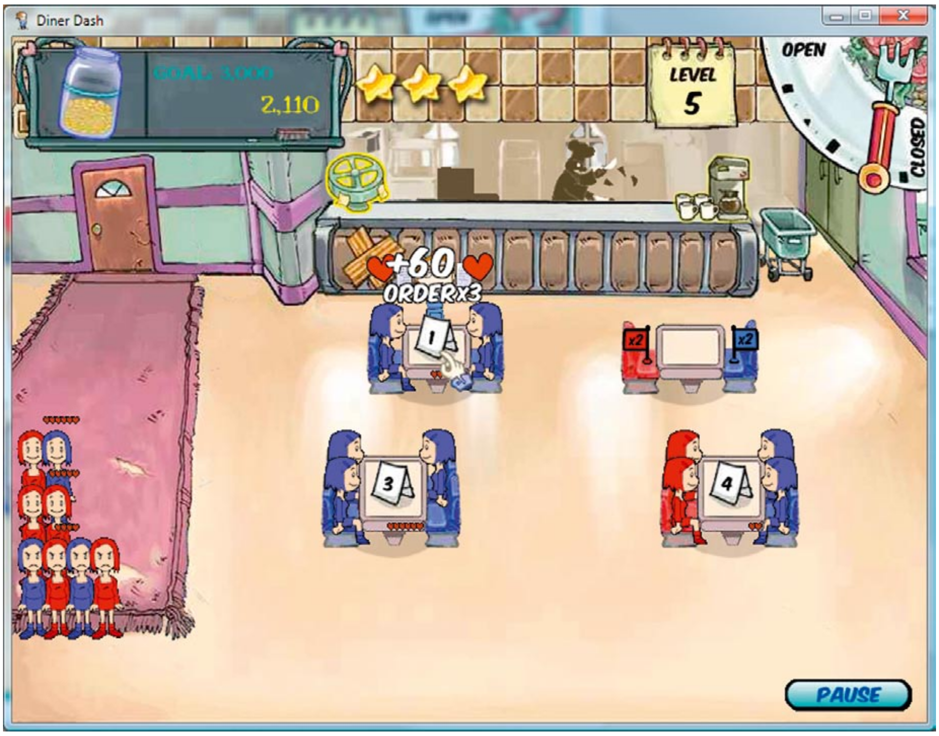
When you perform the same action consecutively, you receive a bonus multiplier for chaining the actions. The more times you perform the action, the longer the chain and the bigger the multiplier. (Copyright © 2003 PlayFirst, Inc. Reproduced by permission of PlayFirst, Inc.)

This incentive to chain puts you at odds with the straightforward play path of just serving customers and answering their needs as they come up. Playing without chaining, you would rush over and take a customer's order as soon as their hand comes up, and then bring the food as soon as it appears in the window. But chaining orders prevents you from playing this way. You must line up actions, delaying taking actions on a customer until you can create a situation where you can perform multiple instances of the same action consecutively.

This forces you to hold off serving some customers even if they are getting impatient and losing hearts. For some players, this can be very hard to stomach. When those fumes of anger appear over a customer's head (Figure 9.5), indicating they are about to lose a heart, your first impulse is to rush over and serve them. You know if you wait too long, the customers will leave and you will lose points. But experience has also taught you that you will score more points if you can hold out and chain your actions. Chaining presents both greater rewards and risks.

You must perform the necessary risk assessment on the fly and decide when to chain to maximize points and when to break the chain in order to save a customer. It seems like a straightforward decision, but sometimes it may actually be more cost-effective (in terms of overall score) to lose a customer than to break a series of chains.

FIGURE 9.5



Chaining puts you at risk of losing customers. In the queue on the left, customers are already losing hearts, with no foreseeable place to seat them. This tension animates the game. (Copyright © 2003 PlayFirst, Inc. Reproduced by permission of PlayFirst, Inc.)

To complicate matters, the game layers in multiple types of chaining. You can chain actions on customers, as well as by matching customer colors to seat colors. Lining up all of these actions can get complicated, and a bit stressful, when those customers' hearts start to disappear.

Initially, the game does not require you to chain. The score thresholds are low enough that you can get by merely serving customers on an ad hoc basis. The game teaches you to chain from the very start, but it does not require it. But as the game progresses, the score thresholds rise, forcing the player to confront the rewards and risks of chaining. This adds tension to the gameplay. Eventually, to pass certain levels, the player must chain. At this point, the game rewrites the dominant strategy of the game. Until this point, the dominant strategy (or in this case, the path of least resistance) was to churn through as many customers as quickly as possible; the new dominant strategy requires you to chain in order to maximize points to win the level. The game now teaches you that the path to success is through chaining.

In later levels, the play shifts yet again and it becomes better to quickly serve customers and ignore chaining. This ever-shifting set of strategies keeps the game fresh. The different vectors created by the spinning plates mechanic and the chaining mechanic give the game designer a number of different levers to pull to create differing experiences from level to level. Nick Fortugno, the game designer behind *Diner Dash*, milks this tension, giving the level progression a feeling of evolution.

Chaining wouldn't work, however, if the player were simply being rewarded for doing things in the same order the main mechanic requires. It would reinforce the main mechanic, not add tension to the game. If the game rewarded you with bonus points for following the path of least resistance, the game would become boring, as it would make the dominant strategy even more obvious, robbing you of any need to make strategic decisions.

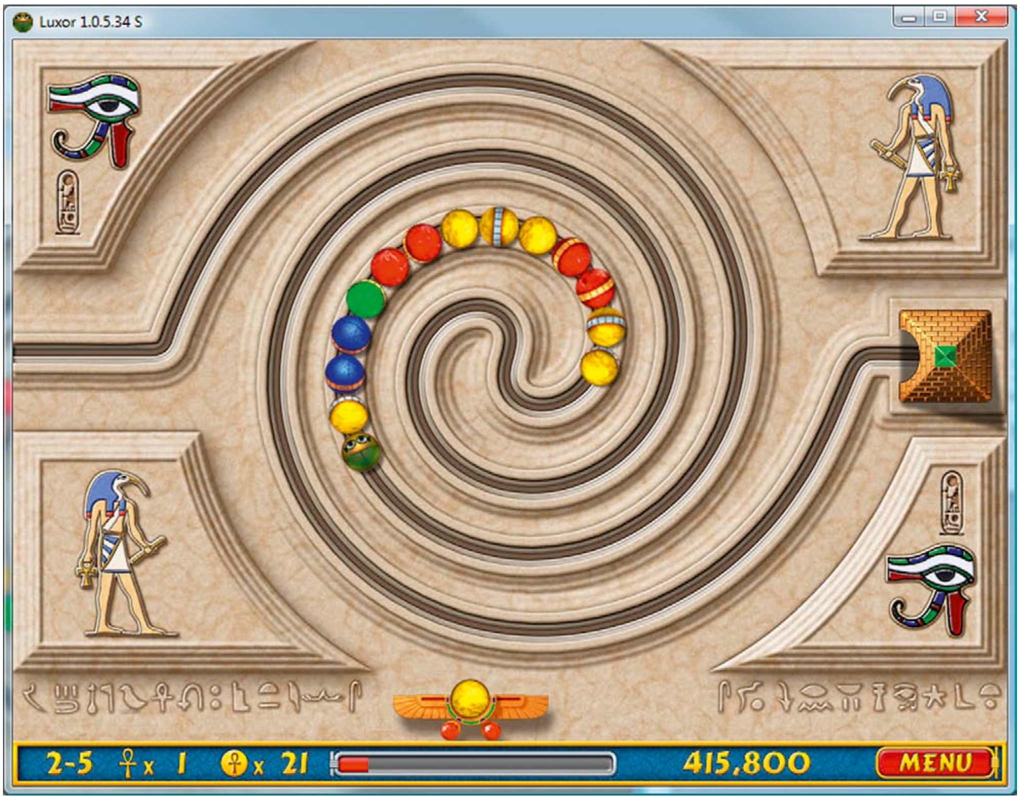
Chaining is a very clear way to shape player actions. The designer can lay out a specific chain of actions, effectively saying, "This is the order I want you to do things in, even if another element of the game says do them in a different order." It also gives the game designer a clear way to say, "I know you want to do this, but you might want to do this." The game designer can lay out these multiple options, then leave it up to you, the player, to choose which path you want to pursue. This gives you a meaningful choice that directly impacts your experience in the game.

Chaining asks you to push your luck to see how long you can accomplish both goals before having to meet the requirements of the main mechanic. The longer you can keep a chain going, the more rewarding (and thus more exciting) the game grows.

Many other games use some form of chaining to help shape the experience. Time management games often use some form of chaining to add tension to the game. We used forms of chaining in *Jojo's Fashion Show* to help shape the levels and add tension. Players scored bonus points in *Jojo's Fashion Show* if they sent out multiple models dressed in the same style in a row. They received an even bigger bonus if they could fully dress all three models and send them out without making changes to their outfits. This incentivized players to quickly dress models and also really consider whether they wanted to change an outfit to give a model a higher scoring top or whether they wanted to take the chaining bonus. Even this small, minor choice complicates play and keeps players on their toes.

Matching games like *Bejeweled* and *Luxor* have their own forms of chaining as well. They enact chaining by rewarding players for multiple matches that occur without having to make other moves. The first set matches and causes other matching pieces to come in contact with one another, creating another match. This encourages players to arrange pieces into arrays that will result in multiple matches when the first match is triggered. To find these chains of matches, you must look several steps ahead. This takes the rather simple mechanic of matching and makes it much more complex as you must account for much more spatial information. By producing these chains, you can reap outsize rewards. But if the game has a time limit or some other game-ending mechanic, setting up the multiple match chains can be risky. Players must quickly weigh the risk and rewards of setting up complex chains of actions.

FIGURE
9.6



Matching games often use a form of chaining seen here in *Luxor*. When you shoot the yellow into the group of three yellow balls in the middle of the loop, the reds on either side will roll together and create another match. (© MumboJumbo)

Chaining certainly isn't the only way to add additional vectors to gameplay. Providing supplementary goals in the form of rewards for specific types of behaviors—such as collecting achievements and items—accomplishes a similar effect. However, they are most effective at creating tension when the supplementary goals don't simply reflect or reinforce the main goal of the game. The dozens, sometimes hundreds, of achievements now built into Xbox games for players to find provide one example of supplementary goals. However, these awards usually serve more to bolster replayability than to add tension to the original play.

Be careful with chaining and additional vectors. Too many additional vectors and the game may grow confusing. The second vector cannot simply run perpendicular to the main gameplay or it will seem tacked on and contradictory. This will confuse players and potentially turn them off. Casual games can have multiple vectors, but they need to be carefully considered. Instead of running perpendicular to the goal of the main mechanic, the secondary mechanic should run at an angle to your core mechanic. It should complicate the achievement of the main goal, but in doing so allow for even greater success than simply following the more obvious path.

Summary

Looking for new ways to build in supplementary goals, chaining and push-your-luck moments into games can be a good way to inject new energy into a lifeless game. For example, you could imagine a *Solitaire* game where the player is rewarded for moving cards of the same color in consecutive moves. This mechanic would stand at odds with the normal alternating color play of *Solitaire*. Some places you might screw yourself and bury a card you need later, but having to make that choice would be a fun and tension-filled moment.

Chaining isn't the only way to add additional vectors to gameplay. It's just a very straightforward one that enables you to directly craft alternate paths. As you design games, you will often find yourself in need of some other vector that pulls the player away from the optimal strategy. The best way to find that other vector is to run through a series of mechanics that pull and push at your original gameplay. It can be nerve-wracking for designers, especially casual game designers looking to make clear and concise games. Additional vectors complicate and cloud the clear core mechanic you spent so long polishing. But you may find great rewards, if you just push your luck a bit.