# Segway System

Narrative System Design Document

Gonçalo de Jesus

# Table of Contents

# Introduction

## Purpose of the Document

The intent of this document is to delineate, analyze, and substantiate the grounds for the creation of a game narrative system with the purpose of addressing a wide lack of natural continuity in game dialogues as a whole.

This system subtly but undeniably sets itself apart from generic, segmented game conversations by introducing a group of phrases and expressions. These are used when prolonging or changing the subject mid-conversation (hereafter referred to as "**segways**"), or to lead back to the plot-critical dialogue (hereafter referred to as "**connectors**"). Optionally, a contextual answer may be provided before moving on with the conversation; these will be referred to as "**responses**".

Example:

*Without the Segway System*

**Player:** Hi!
**NPC:** Hi. There was something Oriana wanted to talk to you about.
**P (choosing non-critical dialogue):** Have you spoken to Dory about the issue she was having with the soldiers?
**N:** Yes, I spoke to her, she said she wanted a word with you when you have a moment.
**P (choosing the critical dialogue, resuming the initial conversation):** What does she want to speak with me about?

*With the Segway System*

**Player:** Hi!
**NPC:** Hi. There was something Oriana wanted to talk to you about.
**P (choosing non-critical dialogue):** Before that: have you spoken to Dory about the issue she was having with the soldiers?
**N:** Yes, I spoke to her, she said she wanted a word with you when you have a moment.
**P (choosing the critical dialogue, resuming the initial conversation):** Alright, thank you. Regarding Oriana, what does she want to speak with me about?

The version including a segway, a response, and a connector has an organic quality to it, a more natural and polished sound. The version without them feels segmented, and even confusing.

Oftentimes, game dialogue is written in sequence; which is fine, but deviations from that sequence (like, for example, selecting two or more non-critical dialogue options before moving forward with the story) end up making the dialogue feel disconnected. Following the example given above, by the time the player hears a “What does *she* want…”, “*she*” can mean many things as the main thread has been lost.

These connections between the main storyline and deviating optional dialogue are what this system aims to address.

## Scope of the System

One of the key aspects of the system is its adaptability regarding scope. If the main objective is to introduce simple segways into conversations, these would add minimal production overhead. Of course, their contribution towards having natural-sounding dialogue would be limited.

Segways, connectors, and responses each individually chip in with their own roles in making lines sound more real and connected. But fortunately, they are pretty modular between themselves, and one of those categories can easily exist without the others. Although their modularity can mean a reduction in production costs in terms of, for example, writing and voice acting, the technical overhead is somewhat constant, due to their shared technical requirements in dialogue tagging and context awareness.

Moreover, its scope will frequently accompany the existing scope in the project it’s being considered for. For example, in a text-based game, the system’s requirements would only add technical and writing overheads. In a voiced game, however, in addition to technical and writing, a voice acting overhead would be introduced. In a fully-fledged 3D action RPG, the segways and connectors could be delivered with variations in intensity, emotion, and many other factors.

Though this is a very rough approximation, it is estimated that there will be a dialogue overhead of around 1% to 4% from adhering to this system, depending on how complex the storyline would already be without it.
As with any other system, its implementation becomes more costly the later it’s introduced into development, so the best practice is to plan ahead for it and start integrating it as soon into production as possible.

A balance must be struck to determine what works best for each project, even after deciding to use the Segway System. As touched upon before, the introduced overhead is relatively fixed, but inside the system, there are a few parameters that can quickly change the values.

| Fixed Technical Overhead | | Variable Technical Overhead | Little to no Technical Overhead, Variable Production Overhead | |
|---|---|---|---|---|
| Dialogue Tagging | Context Awareness | Integration with Existing Systems | Multiple Critical Options | Voice Acting |
| | *(Context Awareness has a fixed technical overhead, and also a variable technical overhead according to the other factors to the right)* | ☛ Were current systems designed with this one in mind?<br>☛ Does the introduction of any of the system's elements break anything?<br>☞ Would waiting for a small sentence break the dialogue system?<br>☞ Is the subtitle system ready for insertion of the elements into the sentences? | ☛ Are the system's elements generic enough to fit all possible paths? | ☛ Is consistent delivery achievable given production requirements?<br>☛ How much can be repurposed from existing lines?<br>☞ And from lines from dialogue not pertaining to the system? |

## Importance of Natural Dialogue Flow

Natural flow in video game dialogue is a vital portion of maintaining player immersion and, to an extent, keeping character voice consistent. For instance, *Dragon Age Inquisition*'s Sera or *NieR Automata*'s pods have unique character voices that benefit from not having connections between thoughts, and thus between conversation themes as well. But apart from these very specific examples – that, at any rate, are meant to be sparsely used –, most characters have much to gain from fluid, real-feeling interchanges.

As agency principles dictate, the player is the pivotal catalyst that drives the discussed theme towards any desired direction in most (if not all) conversations. This makes train-of-thought interconnection all the more important. Non-player characters will always bounce off of what the player selects, so it largely falls upon the playable speaker to introduce these small but effective joints between sentences.

Though this effort is continuous and much has been done in this regard, the Segway System plans to contribute to this everlasting struggle to bring players the most polished narrative experiences possible.

# System Overview

## What are Segways, Connectors, and Responses?

As briefly mentioned before, the system is primarily composed of three components: segways, connectors, and responses.

### Segways

"Segway" refers to a brief phrase, expression, or word that is used to start addressing a new topic, within the same subject as before, or completely changing the conversation theme. Here are two direct examples of what these could be:

*Example 1*

Player: I wanted to ask you a few questions.
NPC: Shoot, chief.
P (selecting a dialogue option that isn't inquiring): We'll get back to that. I need you to take over Phyllis's tasks for the week. She's sick.

*Example 2*

NPC: Hi, *[Player]*! I'm the new mage. I just arrived, which tent should I settle into?
**given three options for tents and one for additional information, the player chooses to obtain more information**
Player (choosing to get more information): Before I answer that, do you prefer spiders or ants inside your sleeping bag?

As seen above, segways' main purpose is ensuring continuity when following non-critical dialogue choices (read *dialogue that doesn't move the conversation forwards*).

### Connectors

"Connector" refers to a brief phrase, expression, or word that is used to return to critical dialogue after a deviation has been made. Its existence avoids situations in which one or more subjects have been introduced between two critical dialogue lines – as the player obtains non-critical information – in such a way that continuing the dialogue naturally would create confusion or lack of clarity.

*Example 1*

NPC: I've been looking for you, *[Player]*! I need you to order the templars to move their dogs. They're driving people away from my tents, and I was already losing enough money with the war as it were.

Player (selecting a non-critical option): What about the recruits? Can't they take care of the dogs?
N: The recruits are trouble enough as it is. They got into a fight with the mages that not even the templars themselves could stop.
P (selecting a critical dialogue option): About the templars, I'll ask them to take care of the dogs. You should have your customers coming back in no time.

*Example 2*

*player character is being picked on in school, and it's about to turn physical; someone the player character doesn't know steps in and drives the bullies away*
NPC: *lends a hand* Get up, my guy. They shouldn't come back any time soon.
Player (selecting a non-critical option): Why'd you help me?
N: I don't really care for bullies, especially those guys.
P (selecting a follow-up non-critical option): You know them?
N: Does anyone *not* know them? Just about everyone here has had their lunch snatched by those douchebags. Oh, they took off with yours. Here's some of mine. *splits a sandwich*
P: Thanks.
N: You got it.
P (returning to critical dialogue): About stepping in, too. Thanks.

Connectors have more of a contextual awareness than segways, but can still be as generic as desired. The ideal way to include them is to make them optional. In the examples above, if only the critical dialogue path had been followed, the connectors would not be needed, as continuity would have been assured by context. But in both cases, there was optional dialogue selected, so the conversation moved away for a bit; when returning to the original line of thought, the connector is needed, otherwise it may become confusing or unnatural.

## Responses

"Response" refers to a brief phrase, expression, or word that is used to bring closure to a side-track dialogue before starting another or returning to the critical dialogue. The main advantage of using responses is to bring closure to a subject before moving on, as one does in everyday conversations.

*Example 1*

Player: Hi, Donna. How are you?
NPC: I'm doing well, sire. A lot of foot traffic around these parts lately. Human trash, if you ask me.
P (selecting a non-critical option): What about that warrior I brought in?
N: That would be Urra. She has been imprisoned, sire.
P (selecting a non-critical option): Thank you. What about that mage we found murdering cattle just off the western fields?
N: That's Orlo, I would think. He's in the last cell to the left, sire.

**P (returning to critical dialogue):** Good. I need to go inside, I bring a letter from Your Majesty Eleanor to bring a prisoner to the town square for trial.

*Example 2*

**Player:** Hi, Elsa.
**NPC:** Hi, *[Player]*.
**P (selecting a non-critical option):** I wanted to ask you about why you joined our crew when we passed you in the woods.
**N:** Well… I suppose you have the right to know. I had been following you for a while, by then.
**P:** Oh?
**N:** You can fight like a demon, but you couldn't notice an elephant tailing you if your life depended on it. Anyways, I'd never really had any friends, but somehow I started to feel like you guys were. I didn't know anyone in your group, but you did the things I didn't have the strength to do by myself. So, I joined ya.
**P (selecting another option, ignoring follow-up options):** I… had no idea. Thank you for telling me. We should be ready to head out of camp soon. When you're ready, head over to Ingrid's tent to pick up your gear.

Responses are the most context-aware elements in the Segway System. In the wider picture, they are usually pre-determined lines at the end of a dialogue line. This is where responses differ from traditional lines that bring a conversation to a close.

Responses are more versatile, and can be placed when the conversation actually moves away from the topic. Analyzing the second example above, at a certain point, the player could have asked follow-up questions. If this had been the case, the line "I… had no idea. Thank you for telling me." would only make sense when changing the subject. If the line was – as is traditional – incorporated into the first part of the conversation, it would have felt less fluid. The player character would be bringing an ongoing conversation to a close, only to reopen it right after.

Another advantage of responses is that they too can be as generic as desired. Although more personalized approaches are preferred, they add more production overhead. Catch-all expressions like "Okay", "Good", or "Thank you" can go a long way to add to the dialogue's flow, at a minimal cost.

## Critical *vs* Non-Critical Dialogue

"Critical dialogue" refers to any dialogue option that moves a conversation forward instead of laterally. To progress in a conversation, a critical dialogue option must be picked; a good example of this is a situation where a character must choose whether their party kills a character or not. The decision would be the critical dialogue path.

On the other hand, asking each party member's opinion about the character's fate would be a lateral move; it would not provide the final choice (as that would fall on the player), it would only provide more information. As such, dialogue options that move a conversation laterally are referred to as “non-critical dialogue”.

## Objectives and Key Features

### Objectives

The Segway System’s main objective is to present an alternative to the widespread norm of segmented dialogue in videogames. It proposes to do so by introducing a series of elements that connect conversation sections together. This is achieved through generic elements whose positions are adaptable, as opposed to adding a connector to each line, as done traditionally.

In other words, instead of having the team record dialogue lines with connectors and placing those connectors manually, the system straightforwardly reads tags in each dialogue line to figure out if there’s any need to include a connector, promptly adding it on the spot if necessary.

In its fully implemented form, the Segway System eliminates the need for excessive variables to keep track of if a player has deviated from the critical dialogue – i.e. dialogue that actually moves the story forward instead of just providing more information. Dialogue tagging is necessary with or without the system, so by adding one or two variables, these conversational transitions can occur seamlessly with reduced technical overhead.

### Key Features

Its core duty is cutting down the need to have particular points in conversations where certain lines are presented. If a player decides to find out more information about a character, something like “back to the task at hand” can be presented when returning to the original conversation topic. There’s no need to check whether the player has returned to the topic, because the system’s tags allow it to understand that the topic has changed, and its contextual awareness will bring up something to connect to the new topic.

With just a small group of dialogue line tags, the system can already obtain the correct connector to present, to make talking with NPCs feel natural and fluid. There is also the added advantage that the system only affects playable characters, as they’ll be the ones driving conversations at any point. This means that the system’s implementation and scope are significantly minimized.

In addition, the system’s nature allows for salvaging portions of existing dialogue lines, or reusing parts of lines that were already planned without the system.

In summary, the system can be as complex as desired, but one of its main characteristics is that a very small amount of effort is needed to get going, and it instantly makes a difference, turning segmented and even unrelated pieces of dialogue into one cohesive and coherent conversation.

## Segway System *vs* Built-In Segways

At first glance, the Segway System may seem equivalent or even interchangeable with traditional dialogue connections, which are built into the dialogue lines themselves. The Segway System’s purpose is to separate these segways, to create a plethora mostly composed of conversation-agnostic transitions that can be taken advantage of at a moment’s notice and in many situations.

A crucial aspect of the system is to facilitate situations where deviations don’t have a defined quantity; which is to say, players can deviate for one sentence, or four or five dialogue options, before returning to the main dialogue. In a traditional, built-in dialogue system, variations of the conversation’s lines would be required to achieve the same effect as the Segway System, because the game would need to actively measure and adapt to the amount of deviation.

This system’s functionalities eliminate the need to play different dialogue lines if the player has deviated from the main conversation, for example. With the introduction of these simple and theme-independent phrases, expressions, and words, the system produces the same effect as built-in transitions, at a fraction of the production cost and with equally noticeable results.

# Design Principles

## Player Agency and Contextual Relevance

One of the Segway System's core pillars is ensuring that context and continuity are at the forefront of the player's experience, regardless of their deviations – and especially then.

To this end, the Segway System's ability to quickly introduce connections between trains-of-thoughts in conversations improves player agency with a detail-oriented contribution. As they go about their interactions with non-playable characters, they feel their choices are incentivized and rewarded with fluid dialogue that makes sense as a whole.

By ensuring that contextual relevance and sensical progression, the player experience is elevated to a much more polished state.

## Realism and Immersion

The game's more polished state allows for more realism; the system enables more genuine social exchanges with NPCs. This in itself increases immersion, but not only that: it propels a more authentic expression of the player character.
For example, with a full integration of the system, it would be possible to have a friendly player character frequently thank their companions, while a hostile or aggressive player character could skip saying "Thank you" altogether, instead opting for "Good" or a similar expression.

These are the types of nuanced aspects that drive immersion forward, and even expand replayability. Different main character personality traits would lead to subtle variations in speech patterns; possibly even driven by player choices throughout the game, as well.

Versatility is the name of the game, here, but the common factor with each addition is the Segway System's potential to strengthen the bond between player and playable character, and to reinforce a game's sophistication and elegance in execution.

# Context Awareness

## Mechanisms for Detecting Context

A critical component of videogame dialogue is the inclusion of tags with each line. This serves a functional need as well as a technical requirement. Designers and writers can take advantage of them to serve as notes about delivery and integration, and programmers can use them to tell the game which line fits where, and what should happen when it's presented to the player.

The Segway System proposes to use this invariably existing system to make its context awareness happen. Lines could, in theory, have as many tags as needed, to ensure that the system's effectiveness is maximized. But to have a quick understanding of how a basic mechanism for the system would work, here are a few examples:

### Subject Tag

The "subject" tag is a core part of the Segway System. It indicates what a line is about, which allows thought connections to directly mention the subject that was originally being discussed, or the one that's just starting to be discussed.

*Connector Example*

| | |
|---|---|
| **Player:** Have you found a solution for the rations? The guards are getting impatient. | *Subject: "Guards"* |
| **NPC:** They haven't eaten in days, but I don't know what I can possibly do. We're running out of supplies across the board. | *Subject: "Guards"* |
| **P (selecting a non-critical option):** What about the spies? Have they found anything in yesterday's scout? | *Subject: "Spies"* |
| **N:** No, sir. They found a few huts to the south, but they'd been ransacked already. | *Subject: "Spies"* |
| **P (selecting a non-critical option):** Could we speak to the General's forces? Maybe they have something they can spare. | *Subject: "General"* |
| **N:** I have, but to no avail. They're dry, same as us. | *Subject: "General"* |
| **P (selecting a critical option):** About the guards, I'll see what I can do. There has to be something... Maybe up north. | *Subject: "Guards"*<br>*(as the subject returns to "Guards", the connector is presented)* |

With just the "subject" tag, the connector can be presented. Although a generic "Moving on" would have already helped the conversation feel more natural, "About the guards" returns to the main talking point seamlessly. By circling back to the original subject with a context-aware connector, the whole exchange feels less segmented, as the transition occurs fluidly.

The "subject" tag can also be used for adding segways as well.

*Segway Example*

**Player:** Have you found a solution for the rations? The guards are getting impatient.

*Subject: "Guards"*

**NPC:** They haven't eaten in days, but I don't know what I can possibly do. We're running out of supplies across the board.

*Subject: "Guards"*

**P (selecting a non-critical option):** We'll get back to that, I'd like to discuss the spies first. Have they found anything in yesterday's scout?

*Subject: "Spies"*
*(as the subject changes to "Spies", the segway is presented)*

**N:** No, sir. They found a few huts to the south, but they'd been ransacked already.

*Subject: "Spies"*

As seen above, the segway is immediately presented on subject change. In truth, this is the core tag of the Segway System, and the base for its fundamental elements to work.

## Recent Tag

There can even be a tag complementary to the "subject" tag to check whether a subject has been touched upon recently; in the example just above, if it were the second time the subject was touched upon, the system would present something like "About the spies again, …" instead of making the full subject change introduction again.

A good rule of thumb would be to reset this flag whenever a conversation starts, and set the subject as recent when it's spoken about. Then, any returns to the subject would present the brief expression instead of a full transition.

*Example*

**Player:** Have you found a solution for the rations? The guards are getting impatient.
**NPC:** They haven't eaten in days, but I don't know what I can possibly do. We're running out of supplies across the board.
**P (selecting a non-critical option):** We'll get back to that, I'd like to discuss the spies first. Have they found anything in yesterday's scout?

*Subject: "Spies"*
*(subject "Spies" becomes recently discussed)*

**N:** No, sir. They found a few huts to the south, but they'd been ransacked already.
**P (selecting a non-critical option):** Before that: Could we speak to the General's forces? Maybe they have something they can spare.
**N:** I have, but to no avail. They're dry, same as us.
**P (selecting a non-critical option):** About the spies again, they should start searching to the east. There may be something there.

*Subject: "Spies"*
*(recently discussed; a briefer segway is more appropriate)*

## Tone Tags

The "tone" tag is situational and would only apply in games where dialogue options can come up in conversations with varying tones (aggressiveness, emotion, hurriedness, among others). In these cases, tone will be a key player in maintaining consistency.

*Examples*

*Situation: rescue mission, the player has arrived at the location where the person in danger was being kept, and is now talking to them.*

**returning to the critical dialogue path after a non-critical conversation that left the player character annoyed (tone: aggressive)**
Player: *brushing the subject off, aggressively* Alright, enough about that. We need to get out of here.

** returning to the critical dialogue path after a non-critical conversation that left the player character nostalgic (tone: emotional)**
Player: *as if saying "enough about us, let's move on"* Alright, enough about that. We need to get out of here.

** returning to the critical dialogue path after a non-critical conversation in a situation where time is of the essence**
Player: *as if wanting to continue the conversation, but not having time to do so* Alright, enough about that. We need to get out of here.

In most cases (as the one above), the tonal differences will mostly belong to the delivery of a line. However, they can just as easily alter the connector itself; for example, the aggressive line above could be omitted entirely and replaced with an annoyed pause of silence, as if to indicate resentment but not a desire to continue the conversation.

"Tone" tags ensure that the conversation's consistency isn't compromised by the introduction of a generic (or specific) element such as a segway or connector, though connectors and responses tend to benefit more from this tag than segways.

## Personality Tags

Certain traits of a player character's personality have an effect on how they respond to things, introduce new subjects, or return to previous ones. For example, a player character with little to no formal education may use more rustic thought connections, while a player character from a noble background will usually be more eloquent and use effective dialogue connections. An impulsive, erratically-thinking player character may even ignore links between conversation themes altogether, because their thoughts aren't exactly connected.

*Examples*

*Situation: the player character is returning to the critical dialogue path after deviating.*

*Player* (simple personality/rustic language): Anyways, back to what we were talking about the crops, is there any way to save them without using up the water reserves?

*Player* (self-assertive personality/eloquent language): Returning to the estate's agricultural produce fields, is there any way to save them without using up the water reserves?

*Player* (erratic personality/unlinked language): Wait, what were we talking about? Right, the crops. Is there any way to save them without using up the water reserves?

A player character's personality is defined by a number of factors; some may be defined in the character creation process, others may even be defined or altered throughout the story according to the player's choices. In any case, it's vital for immersion that their identity is preserved, regardless of the situation. The "personality" tag can be used to ensure this cohesion.

### Pre-Scripted Segway Tags

For the system to allow specific segways, connectors, and responses to be defined, a "pre-scripted" tag is required. In short, before any other segway, connector, or response is considered, this tag is checked. If it's defined, the pre-scripted element is played. If not, the rest of the tags are checked to see what is appropriate to present.

So, in a way, this tag serves as an early exit to not need to check the others, and allows specific answers to be set up. In its purest form, it opposes the general purpose of the Segway System – to provide a dialogue-agnostic solution for transitions between conversation subjects –, but is frequently needed.

## Note Regarding Multiple Dialogue Paths

The Segway System itself will only ever meet the dialogue system's multiple paths in the case of particular segways. Otherwise, connectors and responses (and generic segways) are apart from the dialogue path options. The system identifies where its elements are required through dialogue tagging, which is present in every line, regardless of being in only one dialogue path or multiple.

But if a specific segway should be used to introduce a choice, whereas it wouldn't be used for another choice, that should be explicit to the system. This situation is the only point where the Segway System and the game's general dialogue system directly connect, as previously indicated.

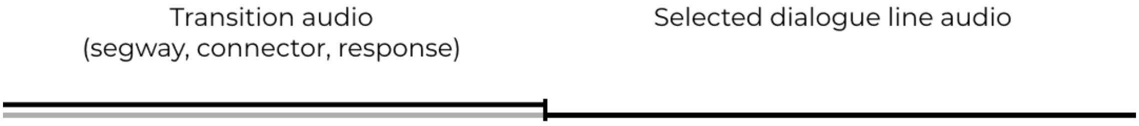
# Implementation

## Dynamic Insertion of Segways

### Audio Integration

After the correct transition has been identified through the tags mentioned in the Context Awareness section, the dynamic insertion of the transition element will require the original dialogue line to wait. In virtually every case, the transition will happen right at the beginning of the dialogue line, so the correct audio should be played there.

The dialogue system should wait until the transition has been played until it plays the selected dialogue line.

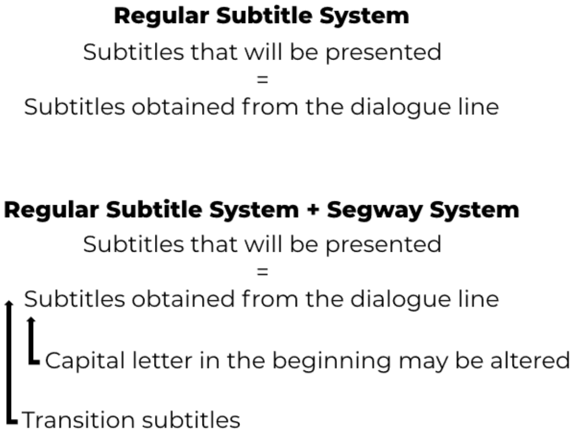


That's it! If the transition has been correctly identified and recording has been consistent, the line should now immediately feel more natural, and the conversation more fluid.

### Subtitle Integration

Similarly to the audio integration, the dialogue line's subtitles will need to be altered on the fly. By now, the transition is known, and its content too. As such, the transition's text can be added to the beginning of the selected dialogue's subtitles when presenting them. This has the advantage of not requiring the subtitles to be changed in the dialogue itself (in the game's files), but rather the subtitle presenter will already integrate the transition before the line's subtitles.

Then, the original dialogue line's first letter may need to be changed. If the transition ends with a comma, for example, the first letter of the selected dialogue line won't be capital, but lowercase.



**Regular Subtitle System**

Subtitles that will be presented
=
Subtitles obtained from the dialogue line

**Regular Subtitle System + Segway System**

Subtitles that will be presented
=
Subtitles obtained from the dialogue line
Capital letter in the beginning may be altered
Transition subtitles

## Existing Dialogue Systems, Requirements and Integration

### Requirements

The Segway System's pre-requisites are relatively simple and have been briefly covered before.

The existing dialogue system must be able to handle the insertion of an audio portion on the fly. By this, it is meant that playing the selected dialogue audio must not be direct. So, in other words, the audio file must be able to be cached in memory, so that one or more audio files can be played before it.

The existing subtitle system must do the same; subtitles must have an intermediate to handle them. So that, instead of directly showing whatever the selected dialogue line's text is, there's a step in-between that allows the insertion of the transition's subtitles as well.

### Integration

The audio file for the selected dialogue line is put on hold; in most game engines (commercial or in-house), there will be one or more audio sources playing the dialogue line, as well as potential animations to be played or camera movements, among many other variables.

When a dialogue option is selected, the selected option cannot be played immediately. The transition checks must occur before, to allow a transition to be presented if necessary. If no transition is necessary, it's business as usual for the dialogue system, which may then directly play the selected option.

The subtitle system must be able to present one batch of subtitles for two separate audiovisual sequences. What this means is that the audio and animations/camera movements for the transition line and the dialogue line will be separate, but the subtitles must not let the player know this. It needs to be a seamless transition.

When the subtitles for the selected dialogue line and its corresponding transition have been obtained, the following happens to correctly present everything:

☛ The dialogue line's subtitles are obtained;
☛ The required transition is identified;
☛ The transition subtitles are copied into a temporary variable, i.e. *tempSubsToPresent*
*(this copy is what will actually be presented)*;
☛ The string with the subtitles is put into another temporary variable, i.e. *dialogueLineSubs;*
☛ *dialogueLineSubs*'s first letter is uppercased or lowercased according to the last character found in *tempSubsToPresent* (i.e. if it ends in a

period, *dialogueLineSubs*'s first character will be capital, if it ends in a comma, *dialogueLineSubs*'s first character will be lowercase.);
☛ *dialogueLineSubs* is inserted into *tempSubsToPresent*
*(usually right at the end, but may be at the start in particular cases)*;
☛ *tempSubsToPresent* is presented to the player, correctly showing the transition and the dialogue line, effectively merged and with the right capitalization.

*Example*

**Dialogue line:** We're got to figure out how to push them away from the stronghold, our forces can't hold them much longer.
**Connector:** About the siege,
**Response:** Alright, I'll tell the scouts when I see them.

>Dialogue option is selected
>Transitions are obtained

>tempSubsToPresent = "Alright, I'll tell the scouts when I see them."

>tempSubsToPresent += " "
>tempSubsToPresent += "About the siege,"
>dialogueLineSubs = "We've got to figure out how to push them away from the stronghold, our forces can't hold them much longer."

>if the last character in tempSubsToPresent ends a sentence (period, exclamation mark, ellipsis, among others), then dialogueLineSubs remains as it is.
if not (comma, semicolon, among others), then the first letter of dialogueLineSubs is changed to lowercase. **This is the case in this example!**

>dialogueLineSubs = "we've got to figure out how to push them away from the stronghold, our forces can't hold them much longer."

>tempSubsToPresent += " "
>tempSubsToPresent += dialogueLineSubs

>tempSubsToPresent is presented to the player as the audio is played

>Result: tempSubsToPresent shows the following (transitions highlighted for illustrative purposes only):

Alright, I'll tell the scouts when I see them. About the siege, we've got to figure out how to push them away from the stronghold, our forces can't hold them much longer.

## Transition Check Order

In a single dialogue line, more than one transition type may be used at a time. Although segways and connectors should never intersect, the general order to add the transition elements is

Responses

Segways

Connectors

By using this order, you ensure that sentences will always make sense. Responses are the loosest of the three types, and should always be loose sentences. Segways and connectors can be either loose sentences or expressions that become part of the dialogue line itself, so they should only be introduced after responses.

As stated, because of their opposing natures, connectors and segways should never clash in terms of priority, as they should never be used at the same time.

# Writing Natural Segways

## Complexity, Genericness, and Character Identity

Segways, responses, and connectors are an innate and integral part of everyday conversations. We go about our lives using them all the time. But in videogames – because of the nature of increasingly complex storytelling branches –, they are often left in the background.

Writing segways that seamlessly connect conversation portions is relatively straightforward when it's considered that segways are no more than that: simple ways to convene a sense of unity between two disparate subjects.

What this means is that there is no need to overcomplicate them. A single "About _____, (…)" can quite quickly steer a dialogue back to the initial subject. In the same way, a "Thank you, I'll see to it soon. (…)" can easily bring closure to a part of the conversation before moving on to another subject.

The most important thing to keep in mind when writing segways is that character voice and identity still matters. As mentioned many times throughout the document, characters with different personalities will use different connections between thoughts. However, this applies to any dialogue piece you write for a specific character/archetype, so it should feel familiar in that sense.

Segways, responses, and connectors, when well written, should diminish the need for specific instances, instead providing a solid arsenal of generic thought connections that – within their own tone, personality, and other factors – sound natural whatever the circumstances.

## Continuity

Generic segways, responses, and connectors should cover continuity between most if not all subjects. The truth is that, even in our everyday lives, seldom is a specific thought connection necessary. This complements the idea that these connections aren't exactly about the subjects themselves, but rather about keeping a conversation flowing.

Their main purpose is to hide (or mask) where the "boundaries" of subjects are within a conversation with an NPC. Should the player select a dialogue option that has nothing to do with the current subject, the exchange's flow should never be interrupted – as it never is in real life.

In situations where continuity must be assured in a particular manner, or a certain sentence is required for gameplay or narrative reasons, specific connections can be set up. The best part is that these connections aren't much different to write than their generic counterparts. For example, if you have a situation in which instead of a catch-all “Thank you, I’ll take care of it.”, you need to mention a specific character to ensure clarity, you can go for “Thank you, I’ll talk to Elena and see what we can do”. The principle is the same, and the specifics are yours to define.

# Recording Transitions (Voice Acting)

## Necessity / “Do we even have to record anything?”

Here’s a bit of potential good news: you may not need to record any segways at all! Depending on the complexity and genericness of your written thought connections, portions of audio can be salvaged from conversations that you already have. This especially applies to games where the Segway system is being retroactively implemented, but can also be applied to games where the Segway System is planned.

In the first, you can use existing audio segments for connections like “Okay”, “Thank you”, and even subjects i.e. in a “Thank you, I’ll see to it. Now, about the templars” connector, you can get these sentences from audio you have already recorded.
Of course, consistency is key, and you’ll need to make sure that tone, personality, delivery, and many other factors are equivalent (or at least compatible). But if everything lines up, you’re good to go, with little to no overhead!

Chances are, though, that even if you have an extensive catalogue of existing dialogue, you’ll need to record a bunch of lines to serve as connections in the Segway System.

If the Segway System is being integrated into a game that is still in its early stages, you can plan ahead to have some of your connections be parts of dialogue you were already going to include in the main dialogue. For example, if you were already going to have the main character thank someone, you can make one of your responses be just “Thank you”, which you can cut out of the original dialogue line to repurpose as a response.

## Maintaining Recording Consistency

As with any audio recorded post-integration, you’ll need to ensure that the conditions are exactly the same, or as close as humanly possible.

If you’re making a large game – namely, one that includes more than one personality or tone –, it’s important to replicate the circumstances in which the original recordings happened. That way, when integrating the audio into the Segway System, you’ll see harmonious transitions that don’t feel like they were implemented afterwards.

It goes without saying, but the recording environment and atmosphere should not only be replicated by the audio crew, but by the voice actor(s) as well. Often, actors need preparation to get into the headspace of the character and the scene. This should also be considered, for the delivery to feel consistent and fluid with the existing in-game audio.

# Considerations and Challenges

## Potential Issues and Solutions

### Delivery Consistency *vs* Scope

As a game introduces more and more personality and tone variations, it expands the Segway System's complexity and overhead proportionally. This has been mentioned in the Scope of the System section.

There isn't exactly a solution for this, as it's a given tradeoff of the system: as you increase the scope of your dialogue and its variety in tone, the Segway System's becomes more complicated too.
Largely depending on the nature of the dialogue lines and the transitions the game requires, some personality and tone deliveries *can* work for multiple situations, but, as mentioned, this isn't a definitive or absolute fix for the scope growth.

## Limitations

### Scope-Reward Tradeoff

Although the Segway System can be applied to any game that includes dialogue – written, spoken, or both –, the truth is that its rewards are barely noticeable in games of smaller scale. Usually, games with audio support for dialogue lines will make the most of the system, and, vitally, games with non-critical choices that move away from the main subject frequently.

In practice, this means that the Segway System produces the best results when integrated into an AAA or AA branching narrative game, or an indie game that is almost solely story-based with choices.

# Final Thoughts

Videogames are one of the most powerful storytelling means available in the modern era. Among may other wonderful achievements, they brought audiovisual life to the already interesting branching stories found in certain book genres.

But frequently, even the most well-written games seem to sometimes be missing a natural touch, something that stands in the way of fluid dialogue, and, consequently, a natural and unified player experience. And more often than not, a huge part of what's missing (if not the entirety) is the connection between thoughts and conversation subjects when talking to NPCs.

It's seldom talked about and addressed in professional and personal reviews, and commonly overlooked by writers, as general continuity is assured by the main dialogue system. It's cast off onto the backlog, and forgotten over time.

But then the game is released, and that *je ne sais quoi* isn't quite there. You're talking to an NPC about one thing, then another, then another, separately. There isn't a unifying glue.

The Segway System aims to fill this gap, and provide players with natural-sounding conversations. Like they would have with real people. This allows immersion to grow to a higher standard than ever before. It allows dialogue to shine, and the writers to shine through them.

The inclusion of the Segway System isn't a matter of functional requirements. But it *is* about making the game *that* much special, giving smaller details their due attention, and making characters really come to life in never-before-seen ways.

Because at the end of the day, what most players will remember about a narrative game is its characters, how they are, how they talk, and how alive they felt.