Welcome to the Haywrites Stardew Valley Dialogue guide!
3/29/2018 Version. 1.0.0

My name is Brendan Hayworth (or haywrites) and I hope this guide will be helpful to you! Please follow me on twitter at https://twitter.com/Haywritesalot.

I am Looking for Narrative Design/Game Design/Writing work in the industry! If you have any leads, or have work on your own game, let me know at haywrites@gmail.com!

Intro

In this guide I will show you not only how to do dialogue for Stardew Valley in a .txt editor, but also how I approach making one of my Narrative Overhauls.

Before you begin, please go to https://drive.google.com/file/d/1jfSbaXVjWXaWxxSMY8XiVVaJk7SUk6A6/view?usp=sharing and download my Starter Kit, which includes the XNB Mod Toolkit, two installers for Notepad ++ for 32 bit or 64 bit computers, and my dialogue season quick reference.

Step One: Finding the character file for editing.

1. In order for us to actually be able to edit the characters, we need to convert them from their original .xnb file into a .yaml file.
2. We can do this thanks to our good friend XNB Mod Toolkit.
3. To find a character you want to edit, go to wherever you installed Stardew Valley, in my case it's through Steam, and go to Content > Characters > Dialogue. (for steam users: C:\Program Files (x86)\Steam\steamapps\common\Stardew Valley\Content\Characters\Dialogue)
4. You will find all the characters names in .xnb form. In this tutorial, let's grab Evelyn.xnb.

Step Two: Turning a .XNB file into a .Yaml file using XNB Mod Toolkit

1. Open the XNB Mod toolkit folder found in your Haywrites Starter Kit.
2. Place the XNB (in this case Evelyn.xnb) inside the PACKED folder.
3. Click on the .bat file in the root called UNPACK FILES.bat.
4. This will unpack the .xnb, and turn it into a .YAML.
5. Open the Unpacked folder, and there is Evelyn.yaml!

Step Two.5: Installing Notepad ++

1. Open your Haywrites Starter Kit.
2. If you're on a 32-bit OS, install the npp.7.5.6.Installer.exe of Notepad++
3. If you're on a 64-bit OS, install the npp.7.5.6.Installer.x64.exe of Notepad++
4. Once installed, you'll be good to go for the next step.

Step Three: Opening .Yaml using Notepad++

Hard Way

1. Open Notepad ++
2. Go to File > Open
3. Go to where your XNB Mod Toolkit Unpacked Folder is
4. Click on the .yaml, in this case Evelyn
5. The .yaml should open up, and you'll see all the text in the file.

Easy Way

1. Open Notepad ++ for the first time.
2. Right click on the .yaml file in the XNB Mod Toolkit Unpacked Folder.
3. Go to Open With…
4. Click on "choose another program/app"
5. Find Notepad ++ in the list, and click "Always open this filetype with this program"
6. In the future, now just double click the .yaml file and it'll open in Notepad ++ no problem.

Step Four: Putting the Season/Heartlevel strings into the .yaml file.

You could technically do this yourself, but this will save a lot of time in the long run.

1. Open your Haywrites Dialogue Starter Kit
2. Open Season Dialogue Reference in Notepad ++
3. Copy everything in there, then paste it in your YAML file under Content.
4. What I do for these mods, in this case Evelyn, is to click at the end of the last line of vanilla dialogue, in Evelyn's case winter_Sat8:, then press enter to make a new line, then ctrl+v.
5. What will happen is, the dialogue will be posted, but you will see certain lines be red.
6. We need them all to be blue. So, on any red line, click right infront of the first letter, hit backspace twice, then enter once.
7. This should make the line blue.
8. Do this for all of the red lines.
9. Why does this happen? I have no clue.

Step Five: Understanding Strings and its structure

spring_Mon8: "" #!String

Part One: The String Itself

The first part of the string is the season, lowercase. So spring, summer, fall, or winter. This tells the game which season to check before loading the dialogue. (spring)

Next is a _ followed by the first three letters of the day of the week, the first being capitalized. So Mon, Tue, Wed, Thu, Fri, Sat, Sun. (spring_Mon)

Then, the heart level the player has with the character, in this case, 8. No space between the day and the heart level. (spring_Mon8)

Then, you need to put a : (colon) (spring_Mon8:)

Then a space, after which put a pair of quotations. (spring_Mon8: "")

Then, put another space and put #!String, which tells the game that is the end of the string. (spring_Mon8: "" #!String)

Part Two: Writing dialogue.

There are four elements that are important to remember when making dialogue, #$b#, #$e#, and $1-9.

Putting @ in a sentence calls up the player name. (Oh, hello @.) becomes (Oh, hello Brendan.)

#$b# Allows the text to continue into a new text box. (I'm not sure how I'm feeling today.#$b#What do you think?)

#$e# is text that is opened after the player finishes the initial conversation. It's how the game is now, before you just can't talk to them anymore. (I'm feeling really fishy.#$e#Anyway, see you.)

$1-9 vary depending on how many portrait faces there are. You place it at the end of your sentence, after your punctuation. (I sure feel happy today.$3) Portraits are in a grid, like so:
XX
XX
XX

This translates numerically like this:
01
23
45

You don't need to put anything in the text to have the neutral face, as it will use it normally no matter what.

Whatever face is in the 1-5(or more) space is what will be used if you use $number.

Each dialogue box can have a face, so you can use different numbers to show the characters emoting. (What a day.$1#$b#I think things will be better tomorrow.$2#$e#We'll have to see.$3)

How to see portraits from the game:

1. Go to your Stardew Valley game folder, then go to Content, then click on Portraits.
2. They are all .xnb files, so you do like you do with all xnbs and use the xnb toolkit to unpack it.
3. What I did is just copy paste all the .xnbs from the Portrait folder and unzip them to have whenever I need them.

So in the end, your string should look like:
spring_Mon8: "What a crazy day!$1#$b#You think everything will work out?$3#$e#I sure hope so.$1" #!String

EXTRA:
There's another piece of syntax for your string that you can use, called _inlaw_partnername
You use this if you want characters to say something about who the player married.
This works for any character, and will only be shown depending on if the player married the person you write down.
So:
winter_Sun10_inlaw_Penny
Or
spring_Mon8_inlaw_Shane

Step Six: Turning your .yaml back into an .XNB

Congrats, your lines are completed.

1. Open up your XNB-Mod-Toolkit
2. Make sure both Packed and Unpacked folders are empty
3. Place the .yaml file in the Unpacked folder.
4. Click PACK FILES.bat
5. In the Packed folder, you should have a .xnb file!

BUT WHAT IF IT DIDN'T WORK
If anything is wrong at all with your syntax in the .yaml, it won't become a .xnb.
That means if there's a weird space somewhere, or you wrote !#String instead of #!String,
Or you forgot to close a " somewhere,
It won't work. It can be madness if you haven't been watching closely.

Here's my solution to trouble shooting such errors:

1. Create a copy of your .yaml file.
2. Go season by season, and see if it can be packed, starting from the top.
3. If spring works, you know there's no errors in spring.

4. If summer doesn't work, you know there's an error in summer somewhere.
5. Then just widdle down until you find each error.

You can also do ctrl+f in Notepad++ and do Replace.
I'll usually put in each of the screwed up spellings to see if they come up.
To list them here: !#String, #e$#, #b$#
Then just replace them with the correct one.

Step Seven: Overwrite the .xnb in the Stardew Directory

1. Take your new .xnb file.
2. Go to Stardew Valley > Content > Characters > Dialogue
3. Drop your .xnb file into the folder, and it'll ask if you want to replace it. Say yes.
4. Your dialogue is now in the game!

If you want to make sure the .xnb worked, or you want to take screenshots of the dialogue, as I do, for whatever purpose, what I do is, if I have a character in year one, and they are in spring, I'll see the date they are in, say spring_17, and then write a bunch of dialogue in my .yaml for that day to see how it looks.

STEP EXTRA: What I don't cover in this guide.

There are a few more aspects of writing dialogue in Stardew Valley that I don't cover in this guide, because they can be problematic, and I still don't have solutions for them.

--

One is dated numbers, such as spring_17, or summer_1. These texts are only said once, during year one, and no more. Too small a window for a mod, in my opinion.

--

Next is choices. The game allows multiple choice questions, and then characters can remember for a few years after the answer you put to that question. I have not been able to find out a reliable way to create them, as it involves creating a save slot for the string, and then having the game call back to that string, and I haven't been able to figure it out. If you know how to do it reliably, let me know.

--

All of my mods are gender neutral, as the characters never call you he or she, but if you want gender to factor in to what the characters say, how you do that is you put a ^. On the left is said if your character is male, on the right is if your character is female. Male^Female.
So it would read: Hey, bro.^Sup, girl.

--

When you open the dialogue, you may see $h, or $a, which is calling the happy, or angry, or sad face for the characters portraits. I find better control using the above number method myself, as not all characters have happy or sad faces to call.

--

Marriage Dialogue: After you marry a character, their .xnb is switched over to a different .xnb called MarriageDialogueCharactername

I have not been able to wrap my head around getting these to work properly, as it seems to gut the characters of their personalities. In my head there should be more dialogue after marriage, not less, but that is the case.

If I can figure out how to push the current system to its max, that would be great. If you HAVE, please let me know.

--
The last aspect I haven't covered is year-based strings. This allows a character to say something in the year that's indicated, only once and never again.
So it would be like, Sat_1, as in when Vincent talks about his Dad, Kent, coming home.
Or, spring_mon8_1, which would call the text if it's spring, monday, character has 8th heart level, and it's year one.
I tried to do this for my Willy mod way back, and it just got messy and didn't work reliably.

Part Two: Haywrites Approach to Writing Dialogue

In this part, I will go into how I write the Narrative Overhauls, what goes into them, and my design philosophy behind them. If you're curious about how to maintain a standard of quality, you might find this interesting!

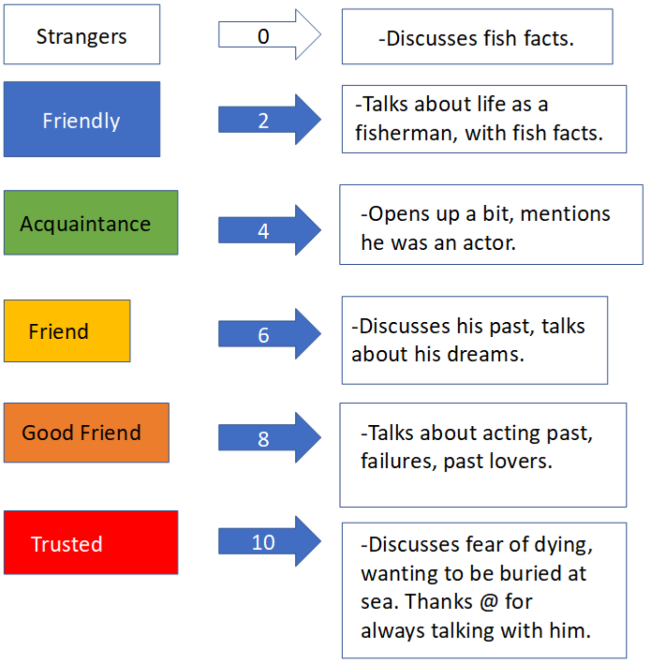
Understanding the Character:

The goal of the Narrative Overhauls is not to reinvent the characters, but to amplify them. To build upon the structure that ConcernedApe built. As such, it's very important that I understand as much as I can about the vanilla character, and to go from there.

What I do is a create a document that I call the Narrative Pitch Guide. Everything they like, or dislike, their current relationships in game, important dates, etc all go into it, so I can see, at a glance how the character could be thinking or feeling at any given time.

Designing the friendship growth:

For me, each heart level is another stage in the friendship. Each character has their own stages, but this is the first one I designed for Willy.



Strangers	0	-Discusses fish facts.
Friendly	2	-Talks about life as a fisherman, with fish facts.
Acquaintance	4	-Opens up a bit, mentions he was an actor.
Friend	6	-Discusses his past, talks about his dreams.
Good Friend	8	-Talks about acting past, failures, past lovers.
Trusted	10	-Discusses fear of dying, wanting to be buried at sea. Thanks @ for always talking with him.

I then take this model and break it down per season. The idea is to have consistency across all seasons because players will be approaching Willy to become his friend from just about any point: from a fresh save, from them installing the mod while they are in year 3 with Willy at 4 hearts, etc.

I wanted people to be able to enjoy the mods and their friendships with the characters from any point.

Writing the Dialogue

Stardew Valley was based on Harvest Moon, and I based the structure on the games themselves. I'm a huge fan of them, and I looked closely at how they wrote their dialogue.

Basically, Harvest Moon dialogue, for me, has always been that there is an implied conversation between the characters, and we only see the most interesting part.

"Hey, farmer." "Hey there, Stu." "How's it been today?" "Pretty good so far." "That's cool." "Yeah." **"A dog almost bit me on the way over here! What's his problem? I only wanted to say hi…"** "Whoa that's crazy."

So when writing my dialogue for Stardew Valley, I'm always focused on capturing these bits to make the dialogue interesting or relevant.

Common Pitfalls

Weather Loop:

It's so damn easy to talk about the weather, and it's super important to not fall into that loop. A very embarrassing story: I found a youtube streamer playing my Marnie Narrative Overhaul. He talked to her over the course of a couple days, and was at the 0 heart level. Marnie ended up talking about how hot it was outside, in different ways, each day. The streamer was like, "She's really bothered by the heat, huh?"

I then realized a few other characters of mine had the same problem, and fixed them in patches. So while talking about the weather is a great way out of feeling stuck, try not to get stuck in it yourself.

No "umms" or "wha?" or "..." in bad places.

For me, what separates someone who is a good from someone who is bad at writing dialogue boils down to how many umms they use. The correct number is zero. Something about that phrase, or any phrase that feels ripped from a bad fanfiction, causes me to lose interest in whatever the character is saying. Same with overused ellipsis (...s) or someone using an ellipsis with only two dots or four dots, and not three.

For me, my ellipsis use is at the end of a sentence, to imply the character trailing off in voice, then at the start of a next text box, to indicate a beat. "I'm not so sure about how I feel today…#$b#...might go home early tonight."

The Linus Conundrum

Last year, I was laid off from my job. I was living off savings while I worked on the mods, but by October I was out of money. I had no food to eat, and was surviving off 20 dollar increments from freelance. Hunger was an everyday thing, and it's amazing how focused you become on wanting to NOT feel hungry when you're in that situation.

So, when I arrived at Linus, as I was living in a similar situation to him, I decided to put everything I was feeling in that moment into his Narrative Overhaul. I released it and...people hated it. It was too sad, too miserable, and not at all Linus.

Their responses were surprising, because everything I had wrote came from an honest place, and I was living so close to Linus it was how I imagined he'd feel. So I did some research, I looked over the vanilla dialogue and found that the common perception of Linus was not that of a down on his luck bum, but more that of a guru.

Recently I went ahead and re-edited the Narrative Overhaul for a second version to match this perception. The critics like it, and it is now far more inline with Linus proper.

However, the lesson I learned was: you can write a character to be miserable...but not TOO miserable.

ENDING

If you have any other questions, or want to share your .yaml with me for some feedback, just e-mail me at Haywrites@gmail.com. I'd love to see what you can do!

End of Guide, 3/29/2018 Version. 1.0.0
Please follow me on twitter at https://twitter.com/Haywritesalot.

I am Looking for Narrative Design/Game Design/Writing work in the industry! If you have any leads, or have work on your own game, let me know at haywrites@gmail.com!