SayiT Best Practices

This document assumes you have some knowledge / practice using SayiT. For initial setup, and explanation of all SayiT's features, please refer to the **SayiT Manual.**

Contents

1 Fit the most readable text possible on the screen

SayiT will *always* perform better if there is **more text** to read available on the screen. With more words to match up, it gives SayiT a higher success rate to hear a word and correctly scroll to it. The following settings contribute to adding more text to the screen on average. Users should assess if the settings will work in their environment and change any settings that they are comfortable changing.

1.1 Anchor Questionnaire

Some of the settings recommended in this document rely on your anchors' comfort level with changing the settings. You'll be adjusting how the text appears on the prompter, and we should take their input when considering what changes can and should be made. We've prepared a survey template for you to use with your anchors to take so you can get an idea of how they feel about some of these changes. Feel free to add any additional questions for different settings that are not listed on the template!

Prompter Adjustment Anchor Questionnaire

1.2 Settings

1.2.1 Limit sluglines to first line only

How to set this in Cue-iT: Prompt tab → Overrides → Slugline Visibility → First Line Only

Sluglines do not contain text that is read – and the full title is usually not important to the talent, while the page number could be.

NOTE: This is especially important if using the "Land on Sluglines" feature. Since sluglines could span multiple lines, "Land on Sluglines" will not function properly if sluglines are completely visible, because there is too high a chance that there will be not enough readable text on screen.

1.2.2 Disable “Force Uppercase” and write scripts in newsroom server with mixed case

How to set this in Cue-iT: Prompt tab → Overrides → Force Uppercase.

We recognize that this is not common, and most studios are still using all uppercase for their font on the prompter. Using mixed case has several significant benefits, so we recommend at least trying it with some of your more open-minded talent!

It is likely that not all talent will be comfortable with this change. The best way to go about this is peace mail – Keep the “Force Uppercase” setting off, write your scripts in mixed case for any talent comfortable with it, and write the script in all caps for any talent uncomfortable with it. Who knows, they may start to prefer the mixed case over time!

<u>Using normal mixed case benefits SayiT in the following ways:</u>

- More text fits on the screen, and on each line
- Mixed case text, in most cases, is ***proven*** to be more legible than uppercase, since it has better visual distinction between words due to varying letter heights and shapes, making it easier for the brain to recognize them quickly
- Easier to spot typos and read words that SayiT will pick up on the fly
- Easier to spot beginning of sentences, and proper nouns which SayiT may not have in its lexicon
- All in all, easier to stick to the script.

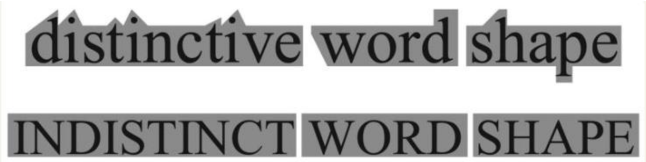
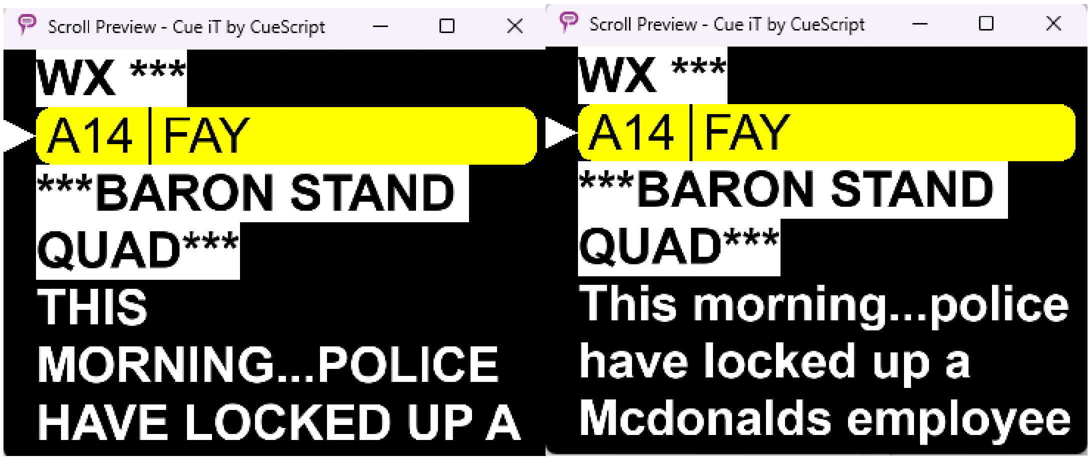


Fig 1. Difference in word shape between mixed case vs. all caps.


Scroll Preview - Cue iT by CueScript
Scroll Preview - Cue iT by CueScript

WX ***
A14 | FAY
***BARON STAND
QUAD***
THIS
MORNING...POLICE
HAVE LOCKED UP A

WX ***
A14 | FAY
***BARON STAND
QUAD***
This morning...police
have locked up a
Mcdonalds employee

Fig 2. Difference in word density when using mixed case on spoken text

1.2.3 Hide blank lines

How to set this in Cue-iT: Prompt tab → Overrides → Hide Blank Lines

Blank lines on the screen mean less readable text on screen, and don’t have much of a use case.

1.2.4 Newsroom Settings – Ignore Stories with empty scripts

How to set this in Cue-iT: Newsroom tab → Connection → Disable → Settings → Options → Ignore stories with empty scripts

With SayiT enabled, Cue-iT will scroll through these empty scripts just fine. However, ignoring these is preferable because it is extra “stuff” that is increasing empty lines and reducing readable text on screen.

1.2.5 Newsroom Settings – Remove tab characters from the script

Tab characters are extra empty space. With a mixed case font, the argument for having tab characters show up on the prompter is worse because the beginning of sentences should be clear.

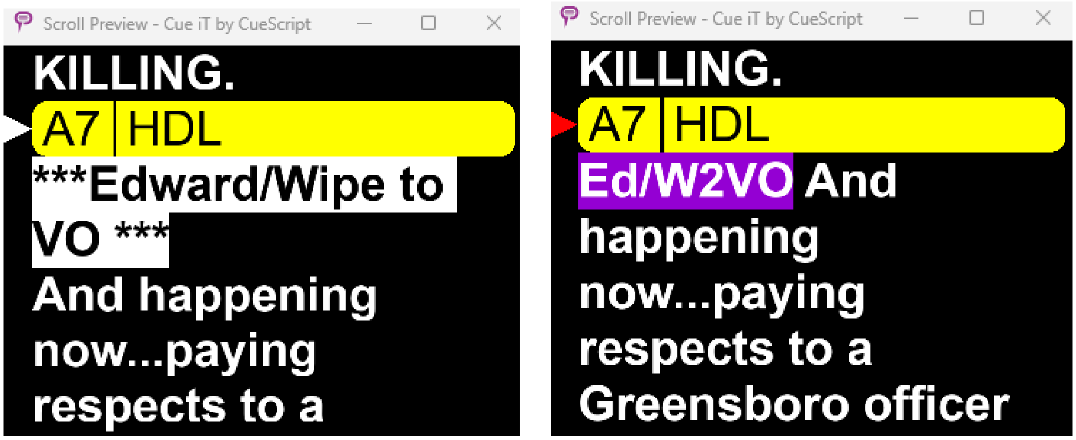
How to set this in Cue-iT: Newsroom tab → Connection → Disable → Settings → Options → Remove tab characters from the script

1.3 Script Formatting

1.3.1 Limit instructions to one line only.

Develop a coding system to trim down names / commands. Include readable text on the same line as instructions if possible. The next graphic has the following changes:

- Remove the extra asterisks, and distinguish this is a production cue by changing its background color
- Shorten Edward to Ed
- Shorten “Wipe to VO” to “W2VO”
- Move the start of the story onto the same line as the instruction if it fits


Scroll Preview - Cue iT by CueScript
Scroll Preview - Cue iT by CueScript

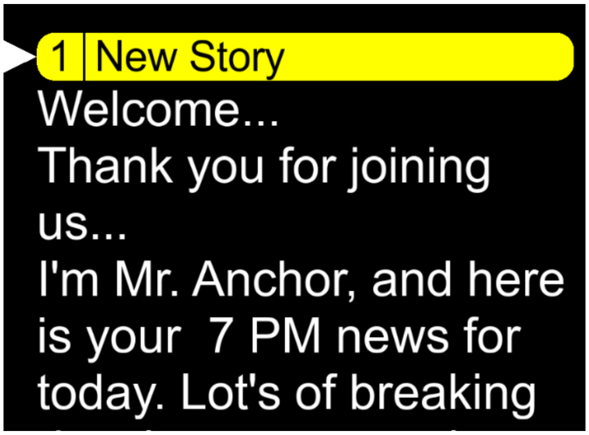
KILLING.
A7 | HDL
***Edward/Wipe to VO ***
And happening now...paying respects to a

KILLING.
A7 | HDL
Ed/W2VO And happening now...paying respects to a Greensboro officer

Fig. 3 Risk has been substantially reduced by trimming the production cue down, 3 lines / 7 words has increased to 5 lines / 9 words

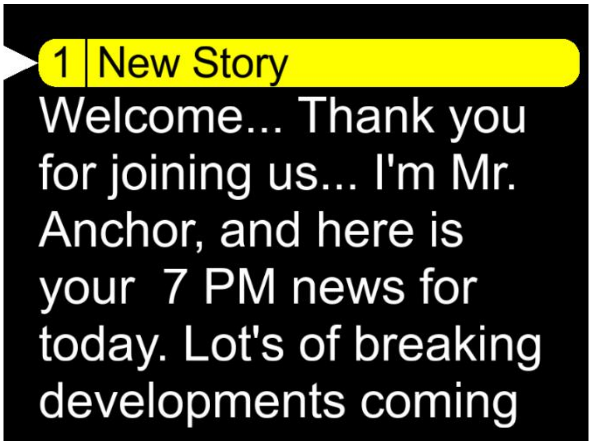
1.3.2 Limit return character usage

Try to avoid using return characters in general or at least limit their usage. They can create a lot of empty space on the prompter, especially if you return after a single word (or happens to return after a single word on a line, like the 2nd example after “us…” in the left screenshot). Again, there’s less use case for using return characters when using mixed case!



1 New Story

Welcome...
Thank you for joining
us...
I'm Mr. Anchor, and here
is your 7 PM news for
today. Lot's of breaking

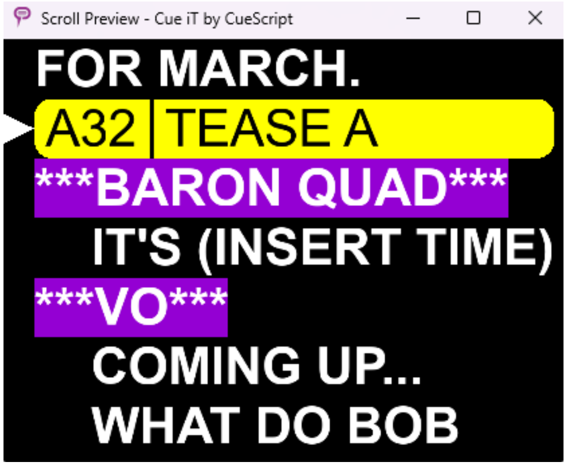


1 New Story

Welcome... Thank you
for joining us... I'm Mr.
Anchor, and here is
your 7 PM news for
today. Lot's of breaking
developments coming

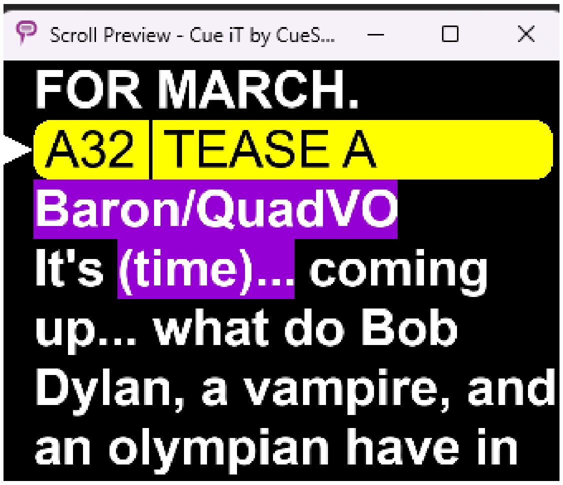
1.3.3 To mark the time and other variables, use instruction text or production cue

Anything on the prompter that does not get explicitly ready should get marked properly so SayiT is not listening for it. The example on the left here is very risky. The example on the right has strategies from this document applied to it and has marked the time properly.



Scroll Preview - Cue iT by CueScript

FOR MARCH.
A32 | TEASE A
BARON QUAD
IT'S (INSERT TIME)
VO
COMING UP...
WHAT DO BOB



Scroll Preview - Cue iT by CueS...

FOR MARCH.
A32 | TEASE A
Baron/QuadVO
It's (time)... coming up... what do Bob Dylan, a vampire, and an olympian have in

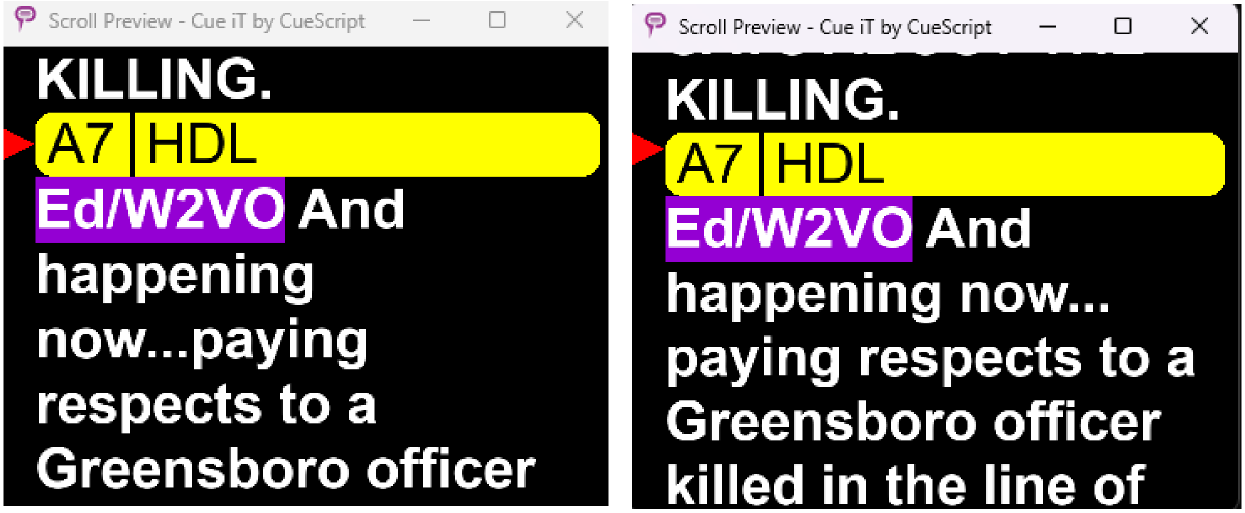
Fig. 4 Risk has been substantially reduced. The two production cues have been merged. Time has been marked properly with a production cue (feel free to mark it with a different color!) Text density has increased greatly as well thanks to mixed case font and removed tab characters.

1.3.4 Always separate words

Always separate wordswith spaces, even when using ellipses, dashes, or other word separators. Replace ellipses with commas if comfortable.

Cue-iT will always smartly create new lines for words that are too long for its current line. It determines what is a "word" by space characters. If you do not include spaces after punctuation, it will assume that the entire string is one word and move it to a new line, which may reduce text density. For example, look at the ellipses in the next graphic.

- "now…paying" is calculated to not fit on the same line as "happening"
- By adding a space and making it "now… paying", Cue-iT can correctly calculate that "now…" can fit on the same line as "happening", further increasing text density by adding a whole new line to the prompter.
- *Additional note: If possible, you can just replace the ellipsis with a comma to note a pause in the sentence. It takes up slightly less space this way and could have a small impact.*


Scroll Preview - Cue iT by CueScript

KILLING.
A7 | HDL
Ed/W2VO And
happening
now...paying
respects to a
Greensboro officer

Scroll Preview - Cue iT by CueScript

KILLING.
A7 | HDL
Ed/W2VO And
happening now...
paying respects to a
Greensboro officer
killed in the line of

Fig. 5 Text density increases further by adding spaces between ellipses. We've increased from 5 lines / 9 words to 5 lines / 14 words. We've effectively doubled the text on screen from the first screenshot in Fig. 3!

1.3.5 Be explicit with numbers

Numbers are tricky for SayiT to handle, they can be said in a variety of ways, especially years and dates. For example:

“1990” could be interpreted as:

> “Nineteen ninety”
>
> “One thousand, nine hundred ninety”

It’s always best to just write out how it’s said.

Write “In the year Nineteen Ninety…” instead of, “In the year 1990”

Write “There have been one thousand, nine hundred ninety accounts…” instead of, “There have been 1990 accounts…”

Write “On the Twenty Seventh of May…” instead of, “On the 27th of May…”

Alternatively, you can add dashes or other punctuation to separate the year numbers so Say-iT is explicitly listening for the correct numbers.

For example, you can write “In the year 19-90", instead of “In the year 1990.”

Importantly, you shouldn’t add dashes for years that are said how they look, for instance the year “2005” should NOT be written as “20-05”, as Say-iT will listen for “twenty-five” or “twenty-oh-five" if you add the dash. Just leave that one as”2005”.

1.3.6 Be explicit with acronyms

When putting acronyms into the prompter, the most consistent way for Say-iT to deal with them is to explicitly mark acronyms with punctuation seperators.

For example, “New FDA mandates are impacting the US”
Should be written as, “New F-D-A mandates are impacting the U-S"

In this example, the punctuation marking for “U-S” is especially important, because Say-iT will be listening for the word “us”, not “U-S”. It’s also potentially confusing for the anchor!

It is especially important to be explicit with acronyms when using ALL CAPS in the prompter, as the sentence would appear as “NEW FDA MANDATES ARE IMPACTING THE US”, you lose the ability to mark acronyms with capital letters.

1.3.7 Use double quotes

When placing quote marks in a script, always use the default double quote marks “”. Double quote marks are stripped when sent to our voice engine, single quote marks are preserved to mark contractions and possessives. For example:

The committee says, “Randy’s performance is unsatisfactory, so we’ve fired him.”

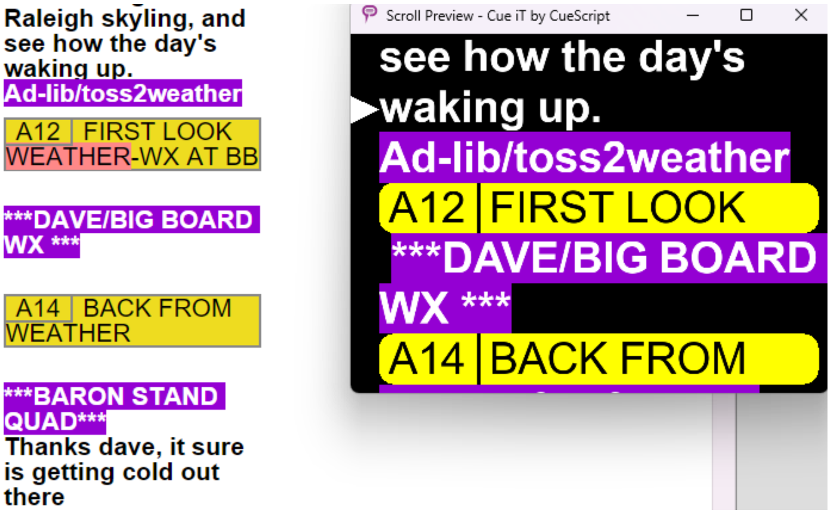
Instead of

The committee says, ‘Randy’s performance is unsatisfactory, so we’ve fired him.’

1.3.8 Adlib instruction recommendation

When instructing the talent to adlib, this should be placed in a production cue or instruction text, so that SayiT is not listening for “adlib to weather”

This can pose a problem if the adlib instruction is placed at the end of a story, since it’s unreadable text, Cue-iT will need to scroll through it to get to the next piece of readable text, and move onto the next story, effectively scrolling through your instruction to adlib.



Raleigh skyling, and see how the day's waking up.
Ad-lib/toss2weather
A12 | FIRST LOOK WEATHER-WX AT BB
***DAVE/BIG BOARD WX ***
A14 | BACK FROM WEATHER
BARON STAND QUAD
Thanks dave, it sure is getting cold out there

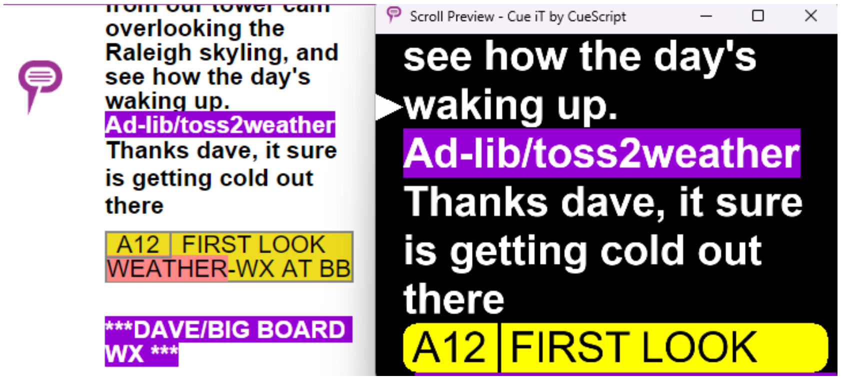
Scroll Preview - Cue iT by CueScript

see how the day's
waking up.
Ad-lib/toss2weather
A12 | FIRST LOOK
***DAVE/BIG BOARD
WX ***
A14 | BACK FROM

Fig. 6 Problem scenario! After finishing “see how the day’s waking up”, Cue-iT will scroll through to A14 and the anchor may miss the instruction to ad lib and toss to weather!

You have two options in this scenario:

1. Keep the next readable text in the same story as the toss. The advantage of this is that no manual control is needed to continue the newscast. However, this forces a re-organization of the stories,



overlooking the Raleigh skyling, and see how the day's waking up.
Ad-lib/toss2weather
Thanks dave, it sure is getting cold out there

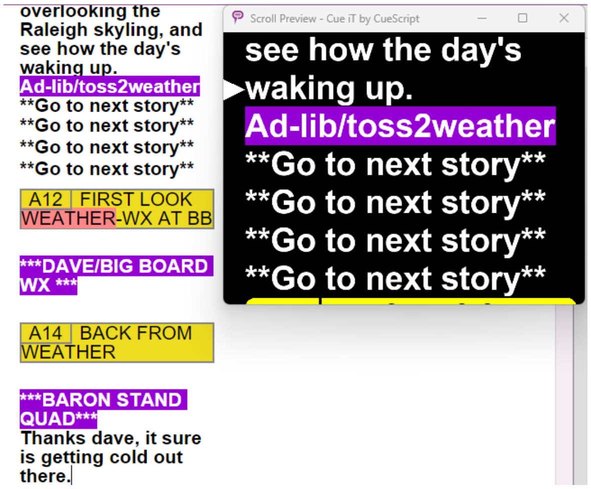
A12 FIRST LOOK WEATHER-WX AT BB

***DAVE/BIG BOARD WX ***

see how the day's waking up.
Ad-lib/toss2weather
Thanks dave, it sure is getting cold out there
A12 FIRST LOOK

Fig. 7 Problem solved! Cue-iT will simply scroll to "Thanks dave" and keep the ad-lib instruction on the screen.

2. Add dummy "readable" lines to instruct a manual "Next Story" button press. Make sure to mark these lines so it's clear they should not actually be read! The advantage of this is you retain your story organization; however, this forces a manual input to be done by some back-up control.



overlooking the Raleigh skyling, and see how the day's waking up.
Ad-lib/toss2weather
Go to next story
Go to next story
Go to next story
Go to next story

A12 FIRST LOOK WEATHER-WX AT BB

***DAVE/BIG BOARD WX ***

A14 BACK FROM WEATHER

BARON STAND QUAD
Thanks dave, it sure is getting cold out there.

see how the day's waking up.
Ad-lib/toss2weather
Go to next story
Go to next story
Go to next story
Go to next story

3.

Fig. 8 Script organization is retained, instruction to go to "Next Story" is clear, and screen will show Adlib instruction on screen after finishing up the previous script

1.3.9 Avoid stories with very little text (less than ~5 lines)

SayiT loads stories one at a time. Stories that contain only a few lines should be avoided for this purpose, a story loaded with only 3 lines of text means we only have 3 lines of readable text on the screen while that story is active, which is risky.

2 Additional considerations

2.1.1 Always have a backup plan

At this point, we have great confidence that Say-iT will follow along very well when your anchors are sticking to the script and annunciating clearly.

However, with live and breaking news, there will be edge cases where Say-iT may need a little extra guidance.

The common edge cases are:

- An anchor is adlibbing and happens to say a phrase within range, which Say-iT hears and causes an unintended scroll
- An anchor does not read what’s on the screen, causing the scroll to stall. Anchors may not read what’s on the screen due to:
  - Poorly written script
  - Because they feel like it !

For this reason, it’s important to have a scroll control back up to guide Say-iT back where it needs to be. Commonly, anchors can either have a foot pedal, or wireless handheld device to scroll the prompter back into position. Cue-iT will smartly disengage Say-iT when taken over by a scroll control, and re-engage after about a half second of stopping the scroll.

When re-engaging Say-iT, it’s best to re-engage when there’s a slight pause in reading. If that’s not possible, try re-engaging the caret exactly where the anchor is reading, or even a line ahead of what the anchor just read. That will give time to re-engage Say-iT, sync back up, and read some words within range to get it back in gear.

2.1.2 Handling on-air updates

On-air updates in general, are not recommended. They can cause the prompter to “jump” if there is a new number of lines given to us from the newsroom, because newsrooms serve Cue-iT the script in one blob, not in an array of lines, Cue-iT has no way of knowing where it should be. Furthermore, it’s dangerous to update text that the anchor may be in the middle of reading!

Cue-iT is now able to support on-air story updates while Say-iT is active, here are our release notes for this feature:

[SayiT] Handle On Air Story Update

SayiT now supports **On-Air story updates** when controlling CueiT during commercial breaks or similar scenarios.

Key Enhancements:

- Ensure the **"Ignore Updates to On-Air story"** feature is **disabled** in the newsroom profile to allow updates.
- If CueiT is positioned on the **Slugline** or the **First Readable Line** of the On-Air story when an update arrives, it will automatically apply the update while maintaining its position.
- If CueiT is actively **scrolling** under SayiT’s control or positioned **beyond the First Readable Line**, the update will be ignored until CueiT moves to the next story.

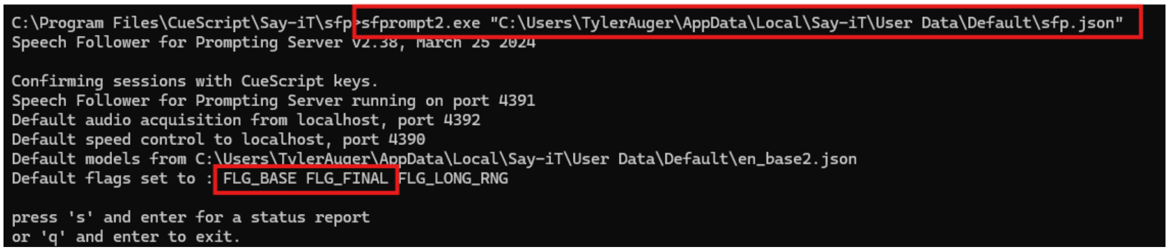
2.1.3 Expand the SayiT lexicon with commonly said unknown names / places / acronyms

SayiT will make its best guess for pronunciation of proper nouns, and it does well for common ones, but sometimes names look nothing like how they’re pronounced. Acronyms are tricky as well, they appear as “words” but are not pronounced as SayiT would think. For this purpose, SayiT comes equipped with the ability to personalize its lexicon. We recommend starting with your anchor names and common acronyms which are likely not recognized and continuing to expand it as necessary (new anchors show up, a particular new place is showing up in the news cycle, etc.).

You can also just replace the names with their pronunciation directly on the output, that may be easier for anchors to deal with in the first place!

The following process will teach you how to see what SayiT is hearing when you come across a name that isn't being recognized. *If you need help with this, contact our support team!*

1. Open up SayiT and disable the process, open up the "Debug Flag" dropdown, select "Show current word" and "Show transcription". Keep SayiT disabled.
2. We'll use the command line to give us a better display of the live console when SayiT is active. To do so, open a command line here: "C:\Program Files\CueScript\SayiT\sfp"
3. You'll then activate SayiT through the command line, and provide your local app data settings file as an argument

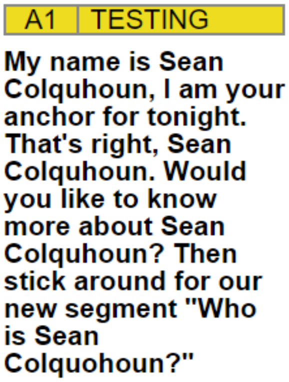


```
C:\Program Files\CueScript\Say-iT\sfp>sfprompt2.exe "C:\Users\TylerAuger\AppData\Local\Say-iT\User Data\Default\sfp.json"
Speech Follower for Prompting Server v2.38, March 25 2024

Confirming sessions with CueScript keys.
Speech Follower for Prompting Server running on port 4391
Default audio acquisition from localhost, port 4392
Default speed control to localhost, port 4390
Default models from C:\Users\TylerAuger\AppData\Local\Say-iT\User Data\Default\en_base2.json
Default flags set to : FLG_BASE FLG_FINAL FLG_LONG_RNG

press 's' and enter for a status report
or 'q' and enter to exit.
```

4. SayiT should now be active in the console.
5. Write up a test story with the problematic name appearing multiple times. In this example, "Sean Colquhon (Shawn Cahun)" is the problematic name

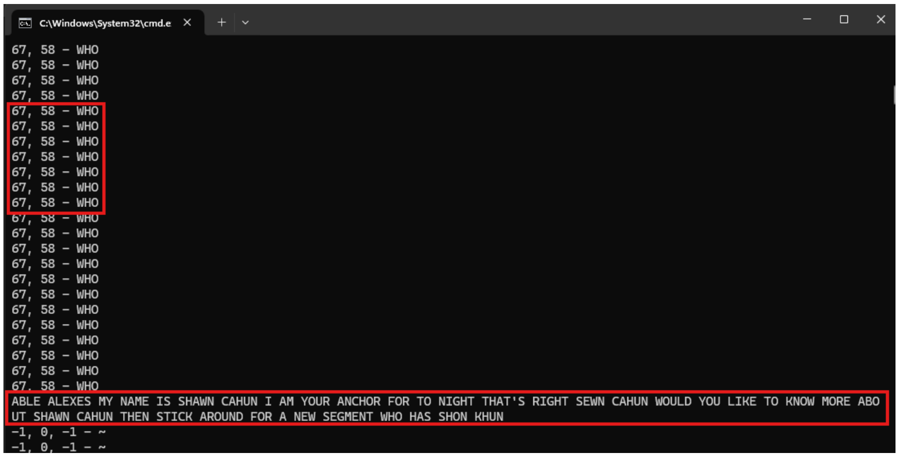


A1 TESTING

My name is Sean Colquhoun, I am your anchor for tonight. That's right, Sean Colquhoun. Would you like to know more about Sean Colquhoun? Then stick around for our new segment "Who is Sean Colquohoun?"

6. In Cue-iT, prompt to this story and enable the SayiT addon.

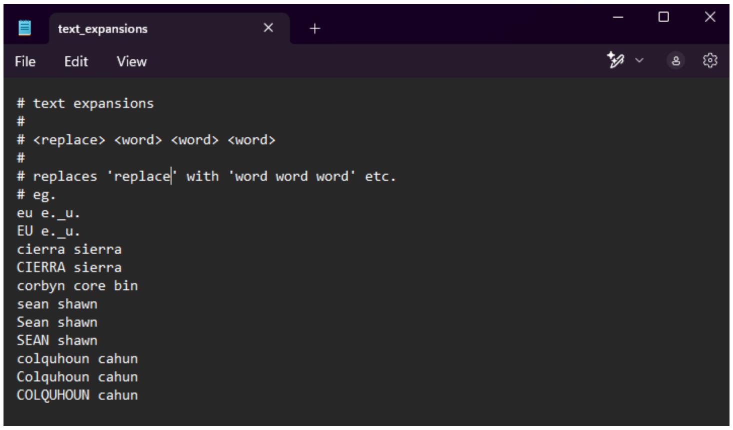
7. Start to read the script and pay attention to your console. The lines with numbers are the last word SayiT heard. If you finish your story, or press next story to exit the current story, a transcript of what SayiT heard in the last story will be logged.



```
67, 58 - WHO
67, 58 - WHO
67, 58 - WHO
67, 58 - WHO
67, 58 - WHO
67, 58 - WHO
67, 58 - WHO
67, 58 - WHO
67, 58 - WHO
67, 58 - WHO
67, 58 - WHO
67, 58 - WHO
67, 58 - WHO
67, 58 - WHO
67, 58 - WHO
67, 58 - WHO
67, 58 - WHO
67, 58 - WHO
67, 58 - WHO
67, 58 - WHO
67, 58 - WHO
67, 58 - WHO
67, 58 - WHO
ABLE ALEXES MY NAME IS SHAWN CAHUN I AM YOUR ANCHOR FOR TO NIGHT THAT'S RIGHT SEWN CAHUN WOULD YOU LIKE TO KNOW MORE ABO
UT SHAWN CAHUN THEN STICK AROUND FOR A NEW SEGMENT WHO HAS SHON KHUN
-1, 0, -1 - ~
-1, 0, -1 - ~
```

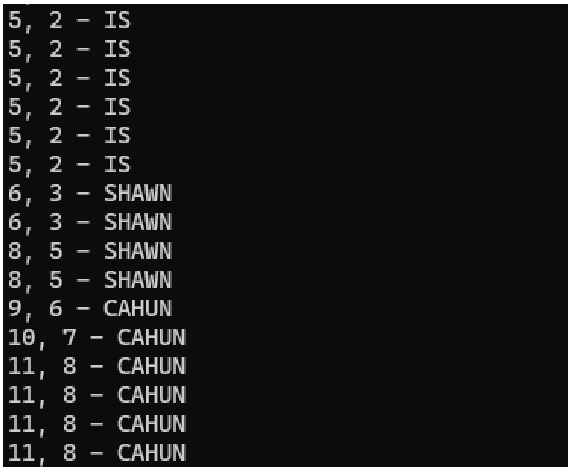
8. If, when you say the name, it's showing up as heard in the console, then we can stop here, SayiT is successfully translating the names pronunciation!
9. If, when you say the name, it's not showing up as heard in the console, then we need to expand the lexicon with what it's hearing in the transcript.
10. In the last image's transcript, we can see "Sean" is most commonly being heard as "Shawn", and Colquhoun is almost always getting heard as "Cahun". Repeat the test as many times as you need to get the most commonly heard pronunciation.
11. Navigate to your language folder within the SayiT program files, for English this is "C:\Program Files\CueScript\SayiT\sfp\en_base2"
12. Show hidden files
13. You may need to add permissions to be able to modify text_expansions (right click -> properties -> Security -> give write permissions for your user).

14. Open text_expansions in notepad. Add the pronunciations. Make sure to add pronunciations for any variation of capital letters as well. *Note the example acronyms as well. Make sure to add these for things like "DC", "USA", etc.*



```
# text expansions
#
# <replace> <word> <word> <word>
#
# replaces 'replace' with 'word word word' etc.
# eg.
eu e._u.
EU e._u.
cierra sierra
CIERRA sierra
corbyn core bin
sean shawn
Sean shawn
SEAN shawn
colquhoun cahun
Colquhoun cahun
COLQUHOUN cahun
```

15. Save the file (make sure not to save as .txt file!), restart the command line process, and try the test again. SayiT should now be recognizing the pronunciation!



```
5, 2 - IS
5, 2 - IS
5, 2 - IS
5, 2 - IS
5, 2 - IS
5, 2 - IS
6, 3 - SHAWN
6, 3 - SHAWN
8, 5 - SHAWN
8, 5 - SHAWN
9, 6 - CAHUN
10, 7 - CAHUN
11, 8 - CAHUN
11, 8 - CAHUN
11, 8 - CAHUN
11, 8 - CAHUN
```

2.1.4 Practice!

Our final note is to remind you that this is *not* a fully automated teleprompter! You are effectively replacing the traditional scroll devices with SayiT and the anchor's voices, and *they* are controlling the scrolling. Have your anchors practice with it, they will learn over time how it works, what reading pace feels good for them, what words they can say to "manipulate" the scroll, and so on.

Contact CueScript

US OFFICE
555 Lordship Blvd

Stratford
CT, 06615
T: +1(203)763-4030

UK OFFICE
Unit 15, First Quarter Business Park
Blenheim Road
Epsom, Surrey, KT19 9QN
T: +44 (0) 20 3757 8880

24 Hour Support: +1(203) 692 4856
support@cuescript.tv
www.cuescript.tv

Ask for TeamViewer Support!