

Conversation Design Principles & Best Practices

Some Theory

Back in 1975, language expert Paul Grice came up with a simple idea about how we talk to each other. He said that people usually want to have clear and effective conversations, and he outlined four basic rules that help us do just that. These rules aren't just for people talking to each other; they can also help us make virtual conversations with computers more human-like.

- **Quality: (Tell the Truth)** Say only what you believe is true. Don't mislead or provide false information.
- **Quantity: (Keep It Just Right)** Share just the right amount of information—no more, no less.
- **Relevance: (Stay on Topic)** Stick to what's relevant to the conversation. Don't stray.
- **Be Clear: (Manner)** Avoid confusion. Explain things in a way that makes sense, and avoid ambiguous statements.

When it comes to interacting with a computerized virtual assistant (IVA), sometimes these rules can be broken. Here's why that might happen, and what we can do to make sure our virtual conversations follow these simple, human rules:

IVA Prompt Example	Maxim Violated	Reason	Fix
Welcome to the Acme Reservation Line. How can I help you today?	Quality	If the VA can only do one or two things, asking “how can I help you?” is false advertising ☹️	Tell the user up front what your VA can do. Always have an intent for “what can you do/help me with?” [DS1]
Can you say the last four digits of your account?	Manner	Ambiguous prompt. What if the user responds “Yes”	“Please provide the last 4 digits of your account”
Are you calling to replace your credit card or your debit card?	Manner	Ambiguous prompt. What if the user responds “Yes” or “Both”	Which would you like to replace: your credit card or your debit card?
Before proceeding further, I need to authenticate you. In order to do that, I will be collecting some details from you...	Quantity	Verbose. Especially in voice, you risk losing your audience attention, interruption, etc.	Let me ask you a few questions to authenticate you. First..
User: Are you open right now? VA: We are open Monday to Wednesday from 8 AM to 5 PM, and Saturday...	Relevance	The VA technically provides an answer to the user, but in a very inefficient manner	“We are currently open and our hours are Monday to Friday, 8 AM to 5 PM.”

Best Practices

Below, we summarize the top best practices to keep in mind when designing conversational experiences.

Design Clear & Consistent Personas

Setting the Personality of Your Virtual Assistant (VA): Why It Matters

Think of your VA like a person, even though it's a “machine.” When people talk to it, they see it as someone with a personality. That's why it's crucial to decide on that personality early on and stick to it. Doing so helps set *clear guidelines* for how the VA should act, and it's a way for businesses to make sure their technology reflects what's important to them and their brand.

What Makes Up a Virtual Assistant's (VA) Personality?

A VA's personality is shaped by two main things: its tone of voice and its characteristics. Here's how they work:

1. **Tone of Voice:** This is how the VA "sounds" to users, and it's the most direct way people will sense its personality. The tone of voice guides how the VA talks, from the words it uses to how it structures sentences. Even if it's a text-based VA, the tone still matters. You can think of the tone on scales like:
 - Casual to Formal
 - Cool to Warm
 - Calm to Excited
 - Terse (brief) to Verbose (wordy)
2. **Characteristics:** These are the traits that define how the VA comes across. They should match the values of the brand it represents. Think of words like Capable, Supportive, or Transparent. These characteristics guide the choice of words, the design, and the overall feel of the conversation.
3. **Voice Interactions:**
 - **Sound & Feel:** Choose a voice that sounds natural and high quality. Avoid overly exaggerated or low-quality voices.
 - **Work with the Customer:** Spend time with the customer to pick the right voice. Test it with common phrases, like greetings, and specific ones, like how the brand's name is pronounced.
4. **Digital & Text-Based Interactions:**
 - **Look & Feel:** Think about colors, fonts, icons, and making it easy for everyone to use. Many customers will have a "brand book" that can guide these choices.
 - **Accessibility Best Practices:** The [W3C Web Content Accessibility Guidelines \(WCAG\) 2.1 Quick Reference](#) is a user-friendly guide designed to help developers and designers understand and implement accessibility standards, ensuring that web content is accessible to individuals with various disabilities.
 - **A Note on Gender:** Many automated assistants use female voices, but this can reinforce stereotypes. *Consider* offering a [range of voices](#) across genders to challenge this practice.

By paying attention to sound, look, and feel, you can create a VA that not only talks in line with the brand but also looks and sounds the part. It's about making the VA feel like a real part of the team, reflecting the brand's values in every way.

In practice, make sure that when you're designing conversations, you're adhering to the VA personality (within reason, not at the expense of good design decisions), and make sure your overall tone is consistent throughout the entire experience.

Understanding Different Ways to Talk: A Guide for Virtual Assistants

If you're designing these conversations, here's what you need to know:

1. **Voice Calls:**
 - **Quick Responses:** People expect quick back-and-forth. Design prompts that help users respond within seconds.
 - **Keep It Simple:** Don't overload users with information. Make it easy for them to answer quickly.
2. **Digital Chats:**
 - **More Time:** Users can take up to 15 minutes to respond, so you can provide more context or links.
 - **Visual Tools:** Use buttons and carousels to make things easy. But make sure they're clear and won't break the conversation.
 - **Longer Prompts:** You can use longer messages, but keep them well-formatted and stick to the rule of providing just the right amount of information.
3. **SMS Conversations:**
 - **A Mix of Voice and Digital:** You can ask more than in voice but less than in digital chats. Be mindful of character limits and the way people text.
 - **Keep It Simple:** Use lists, avoid asking for too much at once, and set clear expectations.
 - **Legal Guidelines:** Be aware of the laws that apply to automated text conversations in your area.
4. **General Tips:**
 - **Align with the Brand:** Make sure the conversation fits the company's values.

- **Consider Different Channels:** Think about how voice, digital, and SMS differ and design accordingly.

Remember, the goal is to make the conversation as easy and intuitive as possible for the user, no matter how they're talking to the virtual assistant. Whether it's a quick voice call or a long text conversation, understanding these differences will help you create a better experience. Here's a nice overview: [📱 SMS Compliance, Texting Laws & Rules - The Complete Guide | SlickText](#)

Setting Clear Expectations and Be Consistent

Designing a virtual assistant (IVA) that works well means making sure users know what it can and can't do. This helps avoid confusion and frustration. Here's how to create a clear and consistent experience:

1. **Define the Scope Early:** Decide what your IVA will do from the start. What tasks will it handle? What will users expect?
2. **Guide the User:** If it's their first time, consider giving a tour of what the IVA can do. Instead of just asking, "How can I help you?" be specific, like, "I can help you check your package status or connect you to an agent."
3. **Include a Help Menu:** Make sure users can easily find answers to questions like, "What can you do?" or "How can you help me?"
4. **Be Clear It's Not Human:** Some virtual voices sound very real. If you use one, make sure to tell users upfront that they're talking to a machine.
5. **Cover Related Tasks:** If the IVA can do one thing, it should do related things too. Like, if it can add items to a shopping cart, it should also let users edit or remove them.
6. **Choose a Voice and Stick to It:** Decide if your IVA will speak as "I" or "we" and be consistent.
7. **Use Consistent Elements:** If you use something like quick replies in one part of the conversation, use them in other similar parts too. Be consistent with how you refer to products or other specific terms.

Remembering the Conversation

Just like in human conversations, a good virtual assistant (IVA) should remember who it's talking to and what's been said. This makes the conversation feel more natural and efficient. Here's how to do it:

- If you know the user's name or have talked to them recently, use that information. For example, "*Welcome back, John! I noticed a recent transaction was denied. Is that why you're calling?*"
- If the user/caller tells you something, remember it for later. If they're booking a hotel in Atlanta, assume they want restaurant recommendations there too. Don't sound like a broken record.
- If you ask a question and don't get an answer, don't repeat the whole thing. Just ask the question again. If you have to repeat instructions, make them shorter the second time.

Pick the appropriate Confirmation Strategy

It is imperative for a virtual assistant to acknowledge that it has *heard* and *understood* what the user has said. So above all else, the best confirmation strategy is to *have* a strategy. Continuing a conversation forward with no notification to the user that we've understood them will elicit feelings of uneasiness at best. At worst, they will not trust that they're on the right path and may interrupt and ask for confirmation that we've heard them correctly.

Establishing user trust is paramount to creating a frictionless experience, and confirmations are one of the best ways to do so. In general, there are two overarching confirmation methods: **Implicit Confirmations**, and **Explicit Confirmations**. Both have a time and place within your conversation design.

Implicit Confirmations are confirmations that are worked into your prompting, while still moving the conversation forward. For example, a user may say: "I would like an update on my claim." The IVA could respond: "Happy to help check the status of your claim. I'll just need to ask you a few questions. First..." In this scenario, it's very clear to the user that the IVA: 1. Understood them correctly, and 2. Is about to start a conversation with them that will lead to the outcome they expect.

In most scenarios, **Implicit Confirmations** work best. They have the advantage of making your conversation sound more "natural", and more importantly, they do not stop the conversation and force a user response. **Implicit Confirmations** can be used to acknowledge an

intent or an entity. Acknowledging an intent should be done in the first message of your intent flow (like the example above) before you begin asking for entities. Entities should be confirmed implicitly as well. For example:

- IVA: *"Are you calling about your home equity loan, or your auto loan?"*
- User: *Auto*
- IVA: *"Thanks, I see that your auto loan has a past due balance of \$49.99. Would you like to make a payment?"*

"Thanks" in the last prompt implicitly acknowledges the user's response, and "auto loan" is included to confirm we've understood the entity.

In general, **Implicit Confirmations** are best used during the "happy path" of your flows. If the conversation gets off track, an error occurs, or the interaction is higher stakes, **Explicit Confirmations** become a better choice.

Explicit Confirmations are hard-stop confirmations where the IVA will read back the information it has understood and ask the user to confirm with either a "yes" "no" or one-step correction. There are a few different scenarios where these kinds of confirmations work best:

1. High stakes or complex interactions. For example, anything involving money should be explicitly confirmed before hitting an API or payment gateway.
2. No match or ASR error scenarios. It's good practice to use an implicit > explicit strategy in ASR error scenarios. For example, take a "user profile lookup" scenario. If an IVA is looking up a user's profile based on a member ID, but no record is found for that ID, the IVA should make sure that it heard the user correctly. If not, after attempting to get the relevant information one more time, the IVA should *take responsibility*, and explicitly confirm if the next attempt was heard correctly.

- IVA: *For your member ID, I have A12345. Is that correct?*
- User: *no*
- IVA: *I'm sorry about that, let's try one more time. What is your member ID?*
- User: *H12345*
- IVA: *That was H12345, is that right?*

Mistakes Will Happen

Here's how to handle them well:

- **Don't Blame the User:** If the virtual assistant (IVA) doesn't understand, it's the IVA's fault. Avoid blaming messages like "it looks like you didn't say your ID correctly." Instead, say something like "I didn't quite get that. What was the ID number?"
- **Ask Specific Questions:** If the IVA doesn't understand, don't just say "I didn't understand." Ask for the specific information you need: "What was the phone number, again?"
- **Use Priming:** Help the user respond by giving examples of how they should answer. For example, "Can you please provide your date of birth? You can say 'September 24th, 1989.'"
- **Know When to Move On:** If you don't get an answer, consider using a default setting or ending the conversation. For example:
 - **[BAD]** *"Sorry. I didn't hear that. Is there anything else I can help with?"*
 - **[GOOD]** *"If there's nothing else you need, I'll let you go. Have a great day and thanks for calling."*
- **Set an Error Threshold:** Decide on a maximum number of errors (usually 2 or 3) before transferring or ending the call.
- **Use Progressive Disclosure:** If the customer insists on multiple attempts, gradually increase the information provided. Start by reframing the question, then offer a touch-tone option (DTMF), and finally a transfer message.

Agent Transfers

When a user wants to speak to a human representative, it's our job to make that transition as smooth as possible. Here's how to do it right:

- **Don't Trap the User:** If a user asks for a representative, don't make it hard for them to reach one. Ask for a brief reason for the call, then transfer them. For example:
 - User: *I want a Representative*
 - IVA: *Sure, I'm happy to transfer your call. What are you calling about today?*
 - User: *Paying My bill*

- IVA: *Thanks. I'm transferring your call now.*
- **Use the Right Agent Queue:** By asking the reason for the call, you can send the caller to the right agent, saving time and frustration.
- **Handle Errors Gracefully:**
 - **No Input:** Don't blame the user for not responding. Offer help, like where to find an ID number.
 - **No NLU Match:** If the IVA doesn't understand, offer context or allow a transfer to an agent.
 - **Lookup Failure:** If the IVA can't find a record, confirm the information and offer a transfer if needed.
 - **Technical/System Errors:** Plan for rare system errors and be ready to transfer to an agent.
- **Be Prepared for Common Requests:** Always be ready for typical requests like "agent," "help," "goodbye," "thanks." Don't let a simple oversight ruin a complex conversation.
- **Provide Answers for Unsupported Requests:** If the IVA can't do something, like product search, have a dedicated response ready. It helps guide the user and avoids generic error messages.
- **Communicate Delays:** If the IVA needs time to look up info, let the user know with a message like "One moment please..." to avoid awkward silence.
- **Avoid Unanswerable Questions:** Don't ask things like "How are you today?" if the IVA can't respond to the user's answer.

Compensate for System Weaknesses

The role of good conversation design in compensating for the weaknesses of AI or the overall system is often underestimated. It serves two main purposes: (A) minimizing the chances that users will follow paths known to cause problems for the system, and (B) addressing and mitigating known system flaws. Based on our past experiences, here are some practical tips to enhance conversation design:

- If your prompt includes both a question and an explanation, place the question at the end. For example, instead of asking, *"How would you like to pay, with a credit or a debit card?"* you might say, *"You can pay by credit or debit. How would you like to pay?"* This approach helps avoid barge-in requests, which can be challenging from an ASR (Automatic Speech Recognition) and Dialog Management perspective.
- ASR, also known as Speech to Text (STT), can struggle with short words like "yes" or "no." If you notice issues, try rephrasing your prompt to move away from yes/no questions.
- ASR often recognizes "yes" more easily than "no." If this is the case, consider rewording your prompt so that the most common response aligns with "yes" rather than "no." For instance, instead of asking, "Is there anything else I can help you with?" where the answer might be "no," you could ask, "Will that be all?" where the response is more likely to be "yes."
- **ASRs and Long Alphanumeric Codes:**
 - If you have a multimodal application, consider having the user type, rather than speak. For example, you may be able to send out a message so that the user can type their code, rather than having to speak it.
 - If multimodality is not an option, and your code is numerical only, you may want to fall back to DTMF after one failed attempt via voice
 - If you have a long alphanumeric code, consider asking for smaller pieces of the ID in smaller steps. For example, if you are dealing with the pattern ABC-123456-AB, you may want to first ask for the first three letters, then for the digit sequence, and then for the last two letters.
- At times, the response that comes out of the Text to Speech engine may sound off in terms of pausing, pronunciation or intonation. While you can play around with the way a word is spelled or adding punctuation to improve how a system prompt sounds, it is best practice to do this programmatically by using SSML syntax within your prompt, which ensures that, for example, you won't have to worry about incorrect spelling if you then need to turn your voice IVA into a chat IVA. Here is the exhaustive list of [ssml tags](#), and here a summary of [the most frequent ones](#), including how to use the International Phonetic Alphabet, how to have the IVA read acronyms correctly, how to add pauses and breaks, how to read dates, phone numbers and foreign words:
- When dealing with processes that require time or are associated with significant latency (more than a 1-second response), it's essential to keep the user informed. If a process takes time, let the user know what's happening by saying something like, "One moment, as I look up/edit/make sure/check..." Unexpected silence should be avoided, as users may start speaking to ensure they're still connected, causing ASR to pick up a signal and the IVA to respond prematurely.

- In chat-based interactions, quick replies are valuable for confirming that the user's input is understood and guiding them in what they can say to the bot. This primes them to provide input that the Virtual Assistant is more likely to comprehend. Depending on the user interface design, you may need to ensure that the exact text of the button or quick reply is included in the training phrases for the relevant intent, creating a more seamless and intuitive user experience.