

The Role of Inclusive Language in Technical Writing

A guide to understanding and implementing inclusive language in technical communication and beyond.

This document defines inclusive language, highlights its importance and impact, outlines core principles, provides real-world examples from GitHub, offers an implementation guide for individuals and organizations, addresses common challenges, showcases real-life implementations by tech companies, and lists additional resources.

Table of contents

Inclusive language 101	4
Definition of inclusive language	4
Importance in technical communication.....	4
Impact on user experience	4
Business impact.....	5
Core principles of inclusive language	6
Person-first approach	6
Gender-neutral language	6
Cultural sensitivity	6
Accessible language.....	6
Avoiding ableist language	7
Age-inclusive terminology	7
Examples of non-inclusive language	8
Master/slave	8
Whitelist/blacklist	8
Sanity check	9
Crippled	9
Implementation guide	11
For individuals writers	11
For organisations.....	11
Common challenges and solutions in implementing inclusive language	13
For individual writers.....	13
For organisations.....	14
Real life examples of implementation	16
Apple	16
Salesforce.....	16
Microsoft	16
Google	16
Additional resources.....	17

Style guides	17
Courses and workshops	17
Forums and communities	18
Training programs.....	18
Tools and software	19
Research and reports	19

Inclusive language 101

Definition of inclusive language

Inclusive language refers to communication that avoids using words, phrases, or tones that might be considered exclusionary or discriminatory, reflecting respect and consideration for all individuals.

Inclusive language is used across various fields, including education, healthcare, business, and technology. It reflects an effort to recognize and respect diversity in all forms, including race, gender, age, ability, and cultural background. Rather than simply avoiding offense, inclusive language promotes belonging and equity by acknowledging people's identities and experiences. As communication practices evolve, inclusive language has become a standard in ethical, professional, and accessible communication.

Importance in technical communication

In technical communication, inclusive language ensures that content is accessible and relatable to a diverse audience. It helps in eliminating biases and potential misunderstandings, allowing for clear and effective dissemination of information.

Impact on user experience

Enhanced readability

By using inclusive language, texts become more universally understandable, increasing engagement and comprehension across different demographics.

Promoting accessibility

Inclusive language ensures that communication is accessible to people with varying backgrounds, abilities, and experiences, making it more universally accessible.

Preventing alienation of users

It avoids making users feel excluded due to language that might be perceived as discriminatory or biased.

Respect and empathy

Inclusive language is seen as a gesture of respect and empathy towards users, which can significantly enhance the user experience. It demonstrates that the organization values diversity and inclusion in its engagement with people.

Business impact

Trust and credibility

Organizations that prioritize inclusive language are perceived as being more equitable and socially responsible, enhancing their reputation.

Broader reach

Companies can engage with a more extensive audience, maximizing their market penetration and effectiveness in communication.

Market accessibility

Organizations that do not adopt inclusive language may miss opportunities to engage with diverse audiences, impacting sales and market expansion.

Talent acquisition

Companies portraying inclusivity are more attractive to a wider pool of talent, enabling them to hire from a more diverse and skilled workforce.



Financial results

While specific financial outcomes from using inclusive language in content are less frequently isolated, there is significant evidence linking diversity and inclusion with better business performance. [McKinsey's 'Diversity Wins' report \(2020\)](#) found that companies more ethically involved in diversity practices are more likely to see higher profits.

Core principles of inclusive language



Person-first approach

This approach emphasizes referring to individuals first, rather than defining them by their disability, disorder, or diagnosis.

 Recommended	 Not recommended
person with dyslexia	dyslexic
person who uses a wheelchair	wheelchair-bound



Gender-neutral language

This involves using terms that do not specify a particular gender when it is unnecessary.

 Recommended	 Not recommended
Chair or Chairperson	Chairman or Chairwoman
firefighter	fireman

Cultural sensitivity

This requires awareness and sensitivity towards different cultural backgrounds. Localization is essential as it can make content accessible to various cultures if done appropriately. This might involve adapting content to fit cultural norms, idioms, and symbols pertinent to specific audiences.

 Recommended	 Not recommended
Japanese people	the Japanese
Native American	Indian

Accessible language



Using clear and straightforward language that everyone can understand, regardless of their reading level or cognitive abilities. This means avoiding jargon and overly complex language that might confuse or alienate.

 Recommended	 Not recommended
---	---

use	utilize
start	initiate



Avoiding ableist language

Ableist language includes terms and phrases that marginalize those with disabilities. Striving for language that is respectful and considers the perspectives and experiences of disabled individuals is crucial.

 Recommended	 Not recommended
ignore	blind to, deaf to
hinder	cripple

Age-inclusive terminology

Language should respect and reflect individuals of all ages. It involves using terms that do not patronize or stereotype individuals based on their age.

 Recommended	 Not recommended
older adults	elderly
legacy status, preexisting	grandfathered

Examples of non-inclusive language

Software development documentation provides a rich source of real-world language usage in technical contexts. To illustrate how inclusive language principles can be applied in practice, this section presents examples drawn from GitHub repositories. GitHub was chosen as a source due to its vast and diverse collection of documentation from various projects and communities worldwide.

As a central hub for software development and collaboration, GitHub hosts millions of repositories containing readme files, comments, and other forms of documentation. This wealth of content reflects the language practices of a broad spectrum of developers and organizations, making it an excellent resource for identifying both inclusive and non-inclusive language patterns. The examples provided here are not meant to single out GitHub or suggest that it is uniquely prone to non-inclusive language. Rather, they serve as practical illustrations of how language choices in technical writing can impact inclusivity, and how more inclusive alternatives can be implemented.

For each example of non-inclusive language identified, a more inclusive alternative will be suggested, demonstrating how small changes in terminology can contribute to a more welcoming and respectful development environment.

Master/slave

While often used to describe a controlling element and a dependent element, these terms recall historical contexts of oppression.

✗ What was found on GitHub:

Guarantee of data consistency among cluster nodes: replacing loss-less semi-sync between MySQL **master** and MySQL **slaves** with Paxos, PhxSQL ensures zero-loss binlogs between **master and slaves** and supports linearizable consistency, which is as strong as that of Zookeeper.

✓ A more inclusive option:

Guarantee of data consistency among cluster nodes: replacing loss-less semi-sync between MySQL **primary** and MySQL **replicas** with Paxos, PhxSQL ensures zero-loss binlogs between **primary and replicas** and supports linearizable consistency, which is as strong as that of Zookeeper.

Whitelist/blacklist

While often used to describe the functionality of blocking or allowing certain items, these terms suggest negative things should be associated with 'black' and positive with 'white,' which can carry racial implications.

✗ What was found on GitHub:

In directory levels you can find preprocessed raw IP lists based on number of **blacklist** occurrences (e.g. levels/3.txt holds IP addresses that can be found on 3 or more **blacklists**).

✓ A more inclusive option:

In directory levels, you can find preprocessed raw IP lists based on the number of **blocklist** occurrences (e.g., levels/3.txt holds IP addresses that can be found on 3 or more **blocklists**).

Sanity check

While often used to describe a quick mental overview, this term can reinforce the notion that individuals with mental health challenges are inferior or flawed.

✗ What was found on GitHub:

modelChecker is a python plug-in written for Autodesk Maya to **sanity check** digital polygon models. It is unopinionated, provides concise reporting, and lets you select your error nodes easily.

✓ A more inclusive option:

modelChecker is a python plug-in written for Autodesk Maya to **validation check** digital polygon models. It is unopinionated, provides concise reporting, and lets you select your error nodes easily.

Crippled

While often used to describe software with limited functionality, this term is an ableist word that tends to be used with a negative connotation describing the disability.

✗ What was found on GitHub:

A sad truth is that most decompilers out there are **crippled**. Many aren't able to decompile trivial constructs, others can't decompile more advanced, those which seemingly can deal with them, are **crippled** by supporting only the boring architectures and OSes.

✓ **A more inclusive option:**

The unfortunate truth is that many decompilers are **limited in functionality**. Some cannot handle simple constructs, while others struggle with advanced ones. Decompilers that manage these complexities often **only support a narrow range** of architectures and operating systems.

✓ ✓ **Or you can go a step further and use plain language principles to improve the text even more:**

Many decompilers **aren't very functional**. Some fail with simple tasks, while others can't manage complex ones. Even those that can handle these tasks often **work only with a limited set** of computer architectures and operating systems.

Implementation guide

For individuals writers

Audit

Conduct a personal audit of your existing documentation to identify non-inclusive language. Tools like Grammarly and Textio can help spot potentially problematic language, allowing you to revise accordingly.

Self-education

Engage in self-guided learning about inclusive language principles. This might include researching real-life examples or case studies to see the impact of language choices.

Practice through writing

Use interactive online workshops or forums to practice language changes and understand their impact in a real-world context. Engaging in discussions can also enhance learning.

Develop personal guidelines

Create a personal style guide that emphasizes inclusive language. This guide should include specific dos and don'ts tailored to your writing style and context.

Seek feedback

Encourage feedback from peers or online writing communities. This can provide diverse perspectives on the inclusivity of your writing.

For organisations

Audit

Perform a comprehensive audit of all existing content, identifying areas that include non-inclusive language. Use tools like Grammarly or Textio to assist in spotting problematic language.

Training programs

Develop and conduct regular training sessions focused on inclusive language principles. These sessions can include real-life scenarios to practice language changes and understand their impact.

Utilize interactive workshops that provide role-playing exercises or discussions to engage participants in learning.

Inclusive language guidelines

Create or update a style guide that includes a section dedicated to inclusive language. This should outline specific dos and don'ts tailored to your organization's context.

Feedback mechanisms

Establish channels such as focus groups or surveys to get feedback from diverse stakeholders regarding the inclusivity of your communications.

Encourage an open-door policy where anyone can raise concerns or suggest changes about language use.

Regular reviews and updates

Implement a schedule for regular review and updating of materials to ensure they remain inclusive as societal norms and language evolve.

Common challenges and solutions in implementing inclusive language

For individual writers

Unfamiliarity with inclusive language

Challenge	Solution
You may not be familiar with which terms are considered non-inclusive or may feel unsure about the current standards for inclusive communication.	Engage in ongoing learning by consulting style guides, attending workshops, and using digital writing tools that highlight non-inclusive language. Referencing examples and recommended alternatives can help build familiarity over time.

Fear of being corrected or criticized

Challenge	Solution
You might hesitate to use inclusive language out of fear of making mistakes or being publicly corrected.	Adopt a growth mindset and treat inclusive language as a skill that improves with practice. Seek constructive feedback from diverse peers and view corrections as opportunities to learn and improve.

Difficulty changing established habits

Challenge	Solution
You may find it challenging to break long-standing communication habits and adopt more inclusive alternatives.	Develop personal guidelines and reminders to reinforce inclusive habits. Practice rewriting sentences using inclusive alternatives to internalize the changes over time.

Inconsistent use

Challenge	Solution
-----------	----------

You may struggle to apply inclusive language consistently across different types of communication.	Create checklists or use editing tools that support bias-free language. Reviewing written content before sharing or publishing can help catch inconsistencies and ensure alignment with inclusive language principles.
--	--

For organisations

Resistance to change

Challenge	Solution
People may be resistant to altering their communication habits or skeptical about the necessity of change.	Implement change management strategies that include education on the benefits of inclusive language, providing clear examples of its impact on business outcomes and workplace culture.

Lack of awareness and understanding

Challenge	Solution
There may be a general lack of understanding about what constitutes inclusive language.	Conduct educational workshops and training sessions that specifically address what inclusive language is and its importance. Provide tangible examples and practical usage scenarios.

Consistent application

Challenge	Solution
Ensuring consistency in the application of inclusive language across all communications and materials.	Develop comprehensive style guides and best practice documents that serve as references. Create a checklist of common non-inclusive terms and their alternatives. Establish a formal review process for important documents and communications. Designate reviewers trained in inclusive language to perform final checks on key materials before they are published or distributed.

Updating existing materials

Challenge	Solution
-----------	----------

The task of reviewing and updating existing materials to align with inclusive language principles can be daunting.	Prioritize materials that are most visible or frequently used, and create a phased plan to update documents gradually. Utilize tools that can aid the transformation, such as writing software with inclusivity settings.
--	---

Fear of making mistakes

Challenge	Solution
Individuals may fear using the wrong terms or being criticized for unintentional errors.	Foster a supportive environment where learning and feedback are encouraged. Encourage open discussions and provide training sessions addressing common mistakes and how to correct them.

Real life examples of implementation



Apple has made significant strides in promoting inclusive language within the tech industry. The company has revamped its coding terminology to remove phrases with historically racist connotations, such as replacing “blacklist” and “whitelist” with “deny list” and “allow list,” respectively. Furthermore, Apple has substituted “master” and “slave” with “primary” and “secondary” to describe device relationships, effectively removing terms that hint at subjugation or control. Their [guide](#) now explicitly calls it “Inclusive Language” instead of “Fair Language,” emphasizing their commitment to inclusivity.



Salesforce has implemented a [comprehensive strategy](#) to integrate inclusive language across all corporate sectors, not just coding. By establishing an advisory board that includes corporate leaders, employees, and members of Employee Resource Groups, Salesforce ensures diverse perspectives are considered. Their Inclusive Product Language team systematically evaluates language decisions, guided by principles such as Representation and Historical Context, to recommend more inclusive alternatives.



Microsoft has actively integrated inclusive language into their communication methods. Their comprehensive [style guide](#) includes a [bias-free section](#) emphasizing gender-neutral and person-first language. Microsoft also integrates inclusive tools within Microsoft Word, which suggest edits to ensure communications remain neutral and respectful, avoiding stereotypes in product naming. They reinforce these principles through employee education programs, encompassing training and practical resources.



Google has been proactive in fostering inclusivity across various aspects of its operations. Beyond their [“Write inclusive documentation” guide](#), the company continually enhances accessibility features in its products. For example, they have improved Google Meet’s live captioning and developed the ChromeVox screen reader for better web navigation accessibility.

Additional resources

Implementing inclusive language practices requires ongoing learning and access to quality resources. The following list of courses, guides, communities, and tools offers valuable starting points for both individual writers and organizations committed to fostering inclusive communication.

Style guides

W3C – Web Content Accessibility Guidelines (WCAG)

While primarily focused on web accessibility, these [guidelines](#) emphasize inclusive language to improve online content accessibility for all users, including those with disabilities.

The Conscious Style Guide

The [guide](#) provides access to a variety of style guides regarding terminology for different communities, alongside articles that debate language usage.

The American Psychological Association (APA) Style

Offers [guidance](#) on bias-free language across different dimensions of identity and background.

Atlassian's Inclusive Language Guide

The [guide](#) includes specific examples and guidelines for avoiding gendered language, recognizing cultural sensitivities, and choosing words with care to foster inclusion.

Microsoft's Writing Style Guide

[The Microsoft Writing Style Guide](#) includes a dedicated section on bias-free language, emphasizing the use of gender-neutral and person-first language.

Google's Developer Documentation Style Guide

[Google's Developer Documentation Style Guide](#) includes the 'Write inclusive documentation' section, which offers best practices for using inclusive language.

Courses and workshops

Coursera: “Gender and Sexuality: Diversity and Inclusion in the Workplace” by University of Pittsburgh

Provides a comprehensive understanding of [gender and sexuality issues in professional settings](#), helping writers develop sensitivity in their language choices.

edX: “Inclusive Leadership” by CatalystX

Offers insights into [creating inclusive environments](#), which can inform more thoughtful and inclusive writing practices.

LinkedIn Learning: “Confronting Bias: Thriving Across Our Differences”

Helps [identify and address personal biases](#), leading to more conscious and inclusive communication.

Forums and communities

WriteTheDocs community

Connects technical writers globally, offering a [platform](#) to discuss inclusive language practices and share experiences.

Technical Writing Forum on Reddit

Provides a [space](#) for peer discussion and advice on inclusive writing in a technical context.

Dev.to’s #inclusivity tag

Offers [insights](#) from developers and tech professionals on inclusivity in tech communication.

Training programs

Diversity, Equity and Inclusion Certificate Program by eCornell

Provides comprehensive [training](#) for organizational leaders to implement inclusive practices, including communication.

Diversity, Equity and Inclusion: DEI Strategies for Business Impact by University of Cambridge

Offers [practical strategies](#) for HR professionals and managers to foster inclusivity, including language use.

Tools and software

Textio

Uses AI to [analyze and improve](#) job descriptions and other corporate communications for inclusivity and effectiveness.

Alex

Provides an easy-to-use, [open-source tool](#) for catching non-inclusive language in various types of writing.

Research and reports

McKinsey & Company's Diversity and Inclusion reports

Offers [data-driven insights](#) into the business impact of diversity and inclusion, helping make the case for inclusive language.

Deloitte's Global Human Capital Trends report

Provides [up-to-date information](#) on global trends in workplace diversity and inclusion, including communication practices.