

7.11 Spelling, Grammar, and Style

[Go To Top](#)

[Return to Naming and File Conventions](#)

The team should write [Repository](#) documentation, comments, file headers, script help text, [Configuration](#) descriptions, generated documentation templates, catalogue entries, and acceptance [Evidence](#) in clear, consistent English.

Project documentation should use UK spelling unless a tool, programming language, external standard, product name, API, command, package, or quoted source requires another spelling. Examples include “artefact”, “behaviour”, “organisation”, “normalise”, “initialise”, “modelling”, and “authorisation”.

The team should use active voice where practical. Documentation should identify the actor responsible for an action, such as “the script validates prerequisites” or “the [Node](#) publishes status”, rather than relying on passive constructions such as “prerequisites are validated” or “status is published”.

The team should use consistent terminology from the architecture documents and the handbook. Terms such as Node, Topic, [Data Structure Definition](#), [Data Structure Instance](#), [Control Plane](#), [Data Plane](#), generated [Artifact](#), [Baseline](#), and acceptance Evidence should keep their defined meanings.

Documentation should use complete sentences, correct grammar, and consistent punctuation. Headings should use consistent capitalisation. Lists should use parallel structure where practical. Short, direct sentences are preferable to long sentences with multiple embedded qualifications.

The team should avoid the following in committed Artifacts:

1. Informal wording that weakens technical clarity.
2. Unexplained abbreviations or local shorthand.
3. Personal pronouns, such as “I”, “we”, “you”, “our”, or “your”, unless they appear in quoted material or an explicitly approved communication Artifact.
4. Passive voice where active voice would make the responsible actor clearer.
5. Personal notes that refer to an individual developer rather than the project Baseline.
6. Temporary comments that do not link to an open issue, risk, or decision.
7. Editing-history labels such as “new”, “old”, “final”, “final-final”, or “backup”.
8. Unresolved comments such as “fix later”, “temporary hack”, “not sure why this works”, or “come back to this”.
9. Machine-specific or developer-specific references, such as local paths, usernames, workstation names, or “Nick’s version”.

Such comments or labels should enter the controlled Baseline only when they appear in an explicitly marked open issue, risk, decision record, or troubleshooting note.

Spelling and grammar checks should form part of the review. Reviewers should correct minor issues when practical and request a revision when unclear writing affects [Traceability](#), reproducibility, reviewability, or Evidence of acceptance.

© 2026 Dido Solutions, Inc. and Jackrabbit Consulting, Inc.

From:
<https://wiki.didosolutions.com/> - **Dido Solutions, Inc Wiki**

Permanent link:
<https://wiki.didosolutions.com/fxdemo/05-part/07-naming-and-file-conventions/07-11-spelling-grammar-and-style/start>

Last update: **2026/08/09 17:26**

