History
John Gruber created the Markdown language in 2004 in collaboration with Aaron Swartz on the syntax,[2][3] with the goal of enabling people “to write using an easy-to-read and easy-to-write plain text format, optionally convert it to structurally valid XHTML (or HTML)”.[4]

Its key design goal is readability – that the language be readable as-is, without looking like it has been marked up with tags or formatting instructions,[9] unlike text formatted with a markup language, such as Rich Text Format (RTF) or HTML, which have obvious tags and formatting instructions. To this end, its main inspiration is the existing conventions for marking up plain text in email, though it also draws from earlier markup languages, notably setext, Textile, and reStructuredText.[9]

Gruber wrote a Perl script, Markdown.pl, which converts marked-up text input to valid, well-formed XHTML or HTML and replaces angle brackets ‘<’ ‘>’ and ampersands ‘&’ with their corresponding character entity references. It can be used as a standalone script, as a plugin for Blosxom or Movable Type, or as a text filter for BBEdit.[4]

Markdown has since been re-implemented by others e.g. in a Perl module available on CPAN (Text::Markdown), and in a variety of other programming languages.[10] It is distributed under a BSD-style license and is included with, or available as a plugin for, several content-management systems.[6][11][12]

Sites like GitHub, Bitbucket, Reddit, Diaspora, Stack Exchange, OpenStreetMap, and SourceForge use variants of Markdown to facilitate discussion between users.[13][14][15][16]

Standardization
Markdown has been characterised by an informal specification[17] and a reference implementation for conversion to HTML. Over time, many Markdown implementations have appeared. People developed these mostly driven by the need for additional features on top of the base syntax—such as tables, footnotes, definition lists (technically HTML description lists), and Markdown inside HTML blocks. The behavior of some of these diverges from the reference implementation. At the same time, a number of ambiguities in the informal specification have attracted attention.[18] These issues spurred an effort by some developers of Markdown parsers for standardisation.

Babelmark[19][20] is a tool available to “[compare] the output of various implementations” to “promote discussion of how and whether certain vague aspects of the markdown spec should be clarified.”[21] Gruber has argued that complete standardization would be mistaken: “Different sites (and people) have different needs. No one syntax would make all happy.”[22]

In March 2016 RFC 7763 and RFC 7764 were published. RFC 7763 introduced MIME type text/markdown with the original variant. RFC 7764 discusses and registered the variants MultiMarkdown, GitHub Flavored Markdown (GFM), Pandoc, CommonMark, and Markdown Extra among others.[23]

History
John Gruber created the Markdown language in 2004 in collaboration with Aaron Swartz on the syntax,[2][3] with the goal of enabling people “to write using an easy-to-read and easy-to-write plain text format, optionally convert it to structurally valid XHTML (or HTML)”.[4]

Its key design goal is readability – that the language be readable as-is, without looking like it has been marked up with tags or formatting instructions,[9] unlike text formatted with a markup language, such as Rich Text Format (RTF) or HTML, which have obvious tags and formatting instructions. To this end, its main

inspiration is the existing conventions for marking up plain text in email, though it also draws from earlier markup languages, notably setext, Textile, and reStructuredText.[9]

Gruber wrote a Perl script, Markdown.pl, which converts marked-up text input to valid, well-formed XHTML or HTML and replaces angle brackets ‘<’ ‘>’ and ampersands ‘&’ with their corresponding character entity references. It can be used as a standalone script, as a plugin for Blosxom or Movable Type, or as a text filter for BBEdit.[4]

Markdown has since been re-implemented by others e.g. in a Perl module available on CPAN (Text::Markdown), and in a variety of other programming languages.[10] It is distributed under a BSD-style license and is included with, or available as a plugin for, several content-management systems.[6][11][12]

Sites like GitHub, Bitbucket, Reddit, Diaspora, Stack Exchange, OpenStreetMap, and SourceForge use variants of Markdown to facilitate discussion between users.[13][14][15][16]

Standardization

Markdown has been characterised by an informal specification[17] and a reference implementation for conversion to HTML. Over time, many Markdown implementations have appeared. People developed these mostly driven by the need for additional features on top of the base syntax—such as tables, footnotes, definition lists (technically HTML description lists), and Markdown inside HTML blocks. The behavior of some of these diverges from the reference implementation. At the same time, a number of ambiguities in the informal specification have attracted attention.[18] These issues spurred an effort by some developers of Markdown parsers for standardisation.

Babelmark[19][20] is a tool available to “[compare] the output of various implementations” to “promote discussion of how and whether certain vague aspects of the markdown spec should be clarified.”[21] Gruber has argued that complete standardization would be mistaken: “Different sites (and people) have different needs. No one syntax would make all happy.”[22]

In March 2016 RFC 7763 and RFC 7764 were published. RFC 7763 introduced MIME type text/markdown with the original variant. RFC 7764 discusses and registered the variants MultiMarkdown, GitHub Flavored Markdown (GFM), Pandoc, CommonMark, and Markdown Extra among others.[23]

History

John Gruber created the Markdown language in 2004 in collaboration with Aaron Swartz on the syntax,[2][3] with the goal of enabling people “to write using an easy-to-read and easy-to-write plain text format, optionally convert it to structurally valid XHTML (or HTML)”.[4]

Its key design goal is readability – that the language be readable as-is, without looking like it has been marked up with tags or formatting instructions,[9] unlike text formatted with a markup language, such as Rich Text Format (RTF) or HTML, which have obvious tags and formatting instructions. To this end, its main inspiration is the existing conventions for marking up plain text in email, though it also draws from earlier markup languages, notably setext, Textile, and reStructuredText.[9]

Gruber wrote a Perl script, Markdown.pl, which converts marked-up text input to valid, well-formed XHTML or HTML and replaces angle brackets ‘<’ ‘>’ and ampersands ‘&’ with their corresponding character entity references. It can be used as a standalone script, as a plugin for Blosxom or Movable Type, or as a text filter for BBEdit.[4]

Markdown has since been re-implemented by others e.g. in a Perl module available on CPAN (Text::Markdown), and in a variety of other programming languages.[10] It is distributed under a BSD-style license and is included with, or available as a plugin for, several content-management systems.[6][11][12]

Sites like GitHub, Bitbucket, Reddit, Diaspora, Stack Exchange, OpenStreetMap, and SourceForge use variants of Markdown to facilitate discussion between users.[13][14][15][16]

Standardization

Markdown has been characterised by an informal specification[17] and a reference implementation for conversion to HTML. Over time, many Markdown implementations have appeared. People developed these mostly driven by the need for additional features on top of the base syntax—such as tables, footnotes, definition lists (technically HTML description lists), and Markdown inside HTML blocks. The behavior of some of these diverges from the reference implementation. At the same time, a number of ambiguities in the informal specification have attracted attention.[18] These issues spurred an effort by some developers of Markdown parsers for standardisation.

Babelmark[19][20] is a tool available to "[compare] the output of various implementations" to "promote discussion of how and whether certain vague aspects of the markdown spec should be clarified."[21] Gruber has argued that complete standardization would be mistaken: "Different sites (and people) have different needs. No one syntax would make all happy."[22]

In March 2016 RFC 7763 and RFC 7764 were published. RFC 7763 introduced MIME type text/markdown with the original variant. RFC 7764 discusses and registered the variants MultiMarkdown, GitHub Flavored Markdown (GFM), Pandoc, CommonMark, and Markdown Extra among others.[23]History
John Gruber created the Markdown language in 2004 in collaboration with Aaron Swartz on the syntax,[2][3] with the goal of enabling people "to write using an easy-to-read and easy-to-write plain text format, optionally convert it to structurally valid XHTML (or HTML)".[4]

Its key design goal is readability – that the language be readable as-is, without looking like it has been marked up with tags or formatting instructions,[9] unlike text formatted with a markup language, such as Rich Text Format (RTF) or HTML, which have obvious tags and formatting instructions. To this end, its main inspiration is the existing conventions for marking up plain text in email, though it also draws from earlier markup languages, notably setext, Textile, and reStructuredText.[9]

Gruber wrote a Perl script, Markdown.pl, which converts marked-up text input to valid, well-formed XHTML or HTML and replaces angle brackets '<' '>' and ampersands '&' with their corresponding character entity references. It can be used as a standalone script, as a plugin for Blosxom or Movable Type, or as a text filter for BBEdit.[4]

Markdown has since been re-implemented by others e.g. in a Perl module available on CPAN (Text::Markdown), and in a variety of other programming languages.[10] It is distributed under a BSD-style license and is included with, or available as a plugin for, several content-management systems.[6][11][12]

Sites like GitHub, Bitbucket, Reddit, Diaspora, Stack Exchange, OpenStreetMap, and SourceForge use variants of Markdown to facilitate discussion between users.[13][14][15][16]

Standardization
Markdown has been characterised by an informal specification[17] and a reference implementation for conversion to HTML. Over time, many Markdown implementations have appeared. People developed these mostly driven by the need for additional features on top of the base syntax—such as tables, footnotes,

definition lists (technically HTML description lists), and Markdown inside HTML blocks. The behavior of some of these diverges from the reference implementation. At the same time, a number of ambiguities in the informal specification have attracted attention.[18] These issues spurred an effort by some developers of Markdown parsers for standardisation.

Babelmark[19][20] is a tool available to "[compare] the output of various implementations" to "promote discussion of how and whether certain vague aspects of the markdown spec should be clarified."[21] Gruber has argued that complete standardization would be mistaken: "Different sites (and people) have different needs. No one syntax would make all happy."[22]

In March 2016 RFC 7763 and RFC 7764 were published. RFC 7763 introduced MIME type text/markdown with the original variant. RFC 7764 discusses and registered the variants MultiMarkdown, GitHub Flavored Markdown (GFM), Pandoc, CommonMark, and Markdown Extra among others.[23]History
John Gruber created the Markdown language in 2004 in collaboration with Aaron Swartz on the syntax,[2][3] with the goal of enabling people "to write using an easy-to-read and easy-to-write plain text format, optionally convert it to structurally valid XHTML (or HTML)".[4]

Its key design goal is readability – that the language be readable as-is, without looking like it has been marked up with tags or formatting instructions,[9] unlike text formatted with a markup language, such as Rich Text Format (RTF) or HTML, which have obvious tags and formatting instructions. To this end, its main inspiration is the existing conventions for marking up plain text in email, though it also draws from earlier markup languages, notably setext, Textile, and reStructuredText.[9]

Gruber wrote a Perl script, Markdown.pl, which converts marked-up text input to valid, well-formed XHTML or HTML and replaces angle brackets '<' '>' and ampersands '&' with their corresponding character entity references. It can be used as a standalone script, as a plugin for Blosxom or Movable Type, or as a text filter for BBEdit.[4]

Markdown has since been re-implemented by others e.g. in a Perl module available on CPAN (Text::Markdown), and in a variety of other programming languages.[10] It is distributed under a BSD-style license and is included with, or available as a plugin for, several content-management systems.[6][11][12]

Sites like GitHub, Bitbucket, Reddit, Diaspora, Stack Exchange, OpenStreetMap, and SourceForge use variants of Markdown to facilitate discussion between users.[13][14][15][16]

Standardization
Markdown has been characterised by an informal specification[17] and a reference implementation for conversion to HTML. Over time, many Markdown implementations have appeared. People developed these mostly driven by the need for additional features on top of the base syntax—such as tables, footnotes, definition lists (technically HTML description lists), and Markdown inside HTML blocks. The behavior of some of these diverges from the reference implementation. At the same time, a number of ambiguities in the informal specification have attracted attention.[18] These issues spurred an effort by some developers of Markdown parsers for standardisation.

Babelmark[19][20] is a tool available to "[compare] the output of various implementations" to "promote discussion of how and whether certain vague aspects of the markdown spec should be clarified."[21] Gruber has argued that complete standardization would be mistaken: "Different sites (and people) have different needs. No one syntax would make all happy."[22]

In March 2016 RFC 7763 and RFC 7764 were published. RFC 7763 introduced MIME type text/markdown with the original variant. RFC 7764 discusses and registered the variants MultiMarkdown, GitHub Flavored Markdown (GFM), Pandoc, CommonMark, and Markdown Extra among others.[23]History
John Gruber created the Markdown language in 2004 in collaboration with Aaron Swartz on the syntax,[2][3] with the goal of enabling people “to write using an easy-to-read and easy-to-write plain text format, optionally convert it to structurally valid XHTML (or HTML)”.[4]

Its key design goal is readability – that the language be readable as-is, without looking like it has been marked up with tags or formatting instructions,[9] unlike text formatted with a markup language, such as Rich Text Format (RTF) or HTML, which have obvious tags and formatting instructions. To this end, its main inspiration is the existing conventions for marking up plain text in email, though it also draws from earlier markup languages, notably setext, Textile, and reStructuredText.[9]

Gruber wrote a Perl script, Markdown.pl, which converts marked-up text input to valid, well-formed XHTML or HTML and replaces angle brackets ‘<’ ‘>’ and ampersands ‘&’ with their corresponding character entity references. It can be used as a standalone script, as a plugin for Blosxom or Movable Type, or as a text filter for BBEdit.[4]

Markdown has since been re-implemented by others e.g. in a Perl module available on CPAN (Text::Markdown), and in a variety of other programming languages.[10] It is distributed under a BSD-style license and is included with, or available as a plugin for, several content-management systems.[6][11][12]

Sites like GitHub, Bitbucket, Reddit, Diaspora, Stack Exchange, OpenStreetMap, and SourceForge use variants of Markdown to facilitate discussion between users.[13][14][15][16]

Standardization
Markdown has been characterised by an informal specification[17] and a reference implementation for conversion to HTML. Over time, many Markdown implementations have appeared. People developed these mostly driven by the need for additional features on top of the base syntax—such as tables, footnotes, definition lists (technically HTML description lists), and Markdown inside HTML blocks. The behavior of some of these diverges from the reference implementation. At the same time, a number of ambiguities in the informal specification have attracted attention.[18] These issues spurred an effort by some developers of Markdown parsers for standardisation.

Babelmark[19][20] is a tool available to “[compare] the output of various implementations” to “promote discussion of how and whether certain vague aspects of the markdown spec should be clarified.”[21] Gruber has argued that complete standardization would be mistaken: “Different sites (and people) have different needs. No one syntax would make all happy.”[22]

In March 2016 RFC 7763 and RFC 7764 were published. RFC 7763 introduced MIME type text/markdown with the original variant. RFC 7764 discusses and registered the variants MultiMarkdown, GitHub Flavored Markdown (GFM), Pandoc, CommonMark, and Markdown Extra among others.[23]History
John Gruber created the Markdown language in 2004 in collaboration with Aaron Swartz on the syntax,[2][3] with the goal of enabling people “to write using an easy-to-read and easy-to-write plain text format, optionally convert it to structurally valid XHTML (or HTML)”.[4]

Its key design goal is readability – that the language be readable as-is, without looking like it has been marked up with tags or formatting instructions,[9] unlike text formatted with a markup language, such as Rich Text Format (RTF) or HTML, which have obvious tags and formatting instructions. To this end, its main

inspiration is the existing conventions for marking up plain text in email, though it also draws from earlier markup languages, notably setext, Textile, and reStructuredText.[9]

Gruber wrote a Perl script, Markdown.pl, which converts marked-up text input to valid, well-formed XHTML or HTML and replaces angle brackets '<' '>' and ampersands '&' with their corresponding character entity references. It can be used as a standalone script, as a plugin for Blosxom or Movable Type, or as a text filter for BBEdit.[4]

Markdown has since been re-implemented by others e.g. in a Perl module available on CPAN (Text::Markdown), and in a variety of other programming languages.[10] It is distributed under a BSD-style license and is included with, or available as a plugin for, several content-management systems.[6][11][12]

Sites like GitHub, Bitbucket, Reddit, Diaspora, Stack Exchange, OpenStreetMap, and SourceForge use variants of Markdown to facilitate discussion between users.[13][14][15][16]

Standardization
Markdown has been characterised by an informal specification[17] and a reference implementation for conversion to HTML. Over time, many Markdown implementations have appeared. People developed these mostly driven by the need for additional features on top of the base syntax—such as tables, footnotes, definition lists (technically HTML description lists), and Markdown inside HTML blocks. The behavior of some of these diverges from the reference implementation. At the same time, a number of ambiguities in the informal specification have attracted attention.[18] These issues spurred an effort by some developers of Markdown parsers for standardisation.

Babelmark[19][20] is a tool available to "[compare] the output of various implementations" to "promote discussion of how and whether certain vague aspects of the markdown spec should be clarified."[21] Gruber has argued that complete standardization would be mistaken: "Different sites (and people) have different needs. No one syntax would make all happy."[22]

In March 2016 RFC 7763 and RFC 7764 were published. RFC 7763 introduced MIME type text/markdown with the original variant. RFC 7764 discusses and registered the variants MultiMarkdown, GitHub Flavored Markdown (GFM), Pandoc, CommonMark, and Markdown Extra among others.[23]