

# The Theia IDE vs VS Code

EclipseSource : 15-19 minutes

The landscape of integrated development environments (IDEs) continues to evolve, offering developers an array of choices tailored to various needs. This article focuses on a comparison between two prominent options: the Theia IDE and Visual Studio Code (VS Code). We'll explore their similarities, differences, and unique features to help you understand which might be the best fit for your development needs.

Before diving into the comparison, it's important to understand some key concepts related to these IDEs. The following information box provides an overview of these concepts for readers who may not be familiar with them:

### Important Concepts

Concept	Description
Theia Project	The open-source initiative by the Eclipse Foundation, encompassing the Theia platform and the Theia IDE. It aims to provide a robust foundation for building modern tools and IDEs.
Theia Platform	A versatile framework for developing custom tools and IDEs, suitable for desktop or cloud environments. Its modularity and extensibility make it applicable for a wide range of applications beyond standard coding environments.
Theia IDE	An out-of-the-box IDE built on the Theia platform, combining flexibility with modern web technologies. It supports the Language Server Protocol (LSP), the Debug Adapter Protocol (DAP), and integrates the Monaco Code Editor.
VS Code	A popular code editor developed by Microsoft, known for its extensive features, ease of use, and rich ecosystem of extensions available through the VS Code Marketplace. It includes some proprietary components and telemetry.
Code OSS	The open-source project behind VS Code, hosting most but not all of its components. Some features exclusive to VS Code are not available in Code OSS, such as certain remote development capabilities and Live Share.

Please also refer to this [comparison between Eclipse Theia and Code OSS](#).

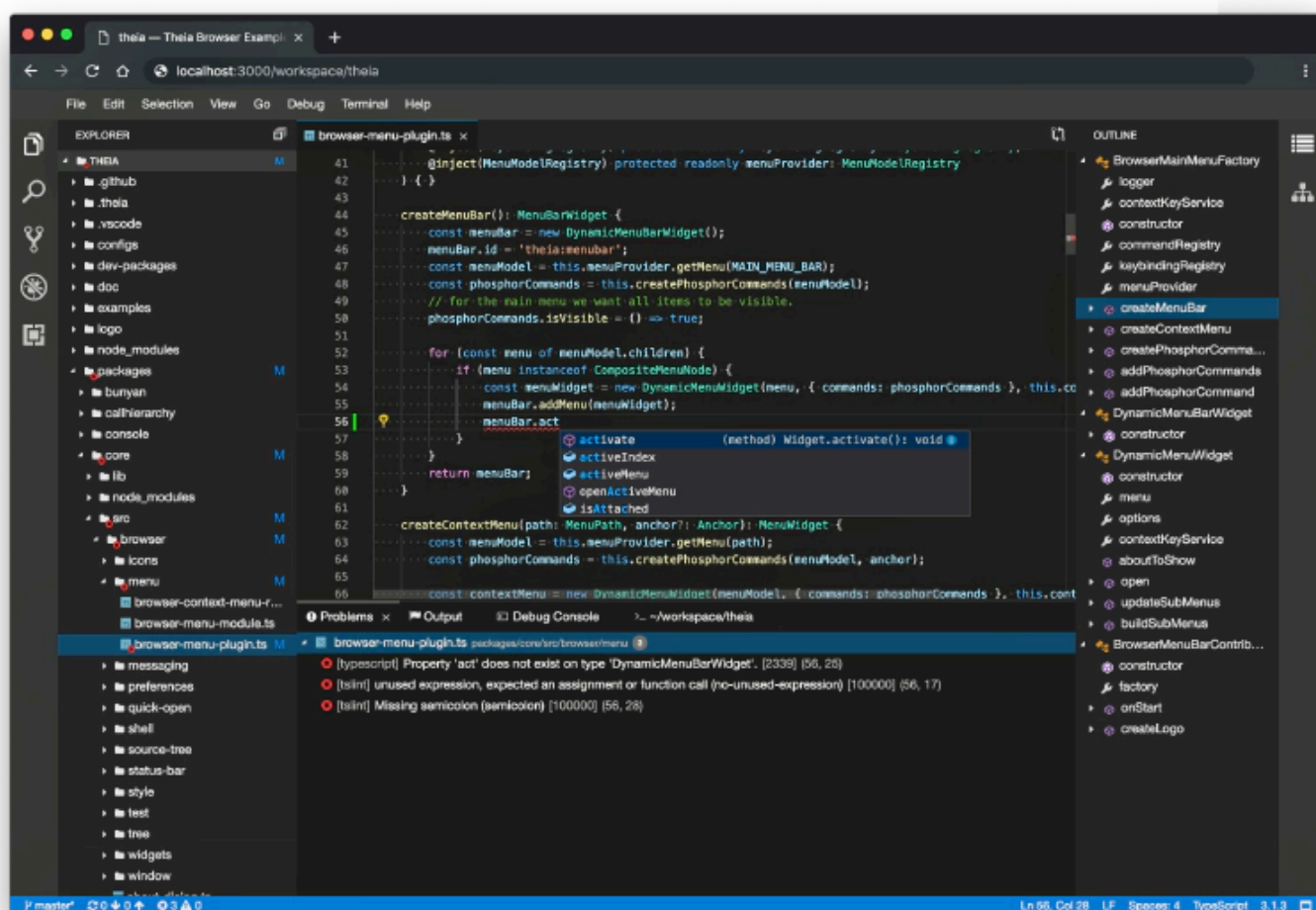
Now, let's focus on a comparison between the Theia IDE and VS Code.

In this comparison, we strive to be neutral, though it is challenging to guarantee complete impartiality. It is a fact that VS Code is much more well-known and has been on the market much longer than the Theia IDE, which has just recently been released. Therefore, in the comparison, we often focus more on the Theia IDE instead of explaining the advantages of VS Code, which we expect are well known to most readers. At EclipseSource, we use, deploy, and extend both tools in various customer scenarios. We strongly believe that every project and stakeholder should make an educated decision and select the ideal solution for their use case. With the recent Theia IDE release, the ecosystem is in a good position to provide more diversity and actually give users another alternative to VS Code.

## Similarities Between Theia IDE and VS Code

Before diving into their differences, it is important to acknowledge the substantial similarities between Theia IDE and VS Code:

- 1. Monaco Code Editor:** Both Theia IDE and VS Code use the Monaco Code Editor, which provides a powerful and versatile code editing experience.
- 2. Language Server Protocol (LSP):** Both IDEs support LSP, enabling consistent and efficient language support, which is crucial for modern development environments.
- 3. Debug Adapter Protocol (DAP):** DAP is also supported by both Theia IDE and VS Code, providing robust debugging capabilities that developers rely on.
- 4. VS Code Extensions:** Theia IDE is compatible with VS Code extensions through the Open VSX registry, allowing users to enhance their development environment from the huge ecosystem of available VS Code extensions.
- 5. User Experience Concept:** Both Theia IDE and VS Code prioritize a streamlined and efficient user experience. Theia IDE, by default, mirrors many of the UX concepts found in VS Code but offers greater flexibility for customization to meet specific needs, e.g. via a user-customizable toolbar and how views can be layouted.



The Theia IDE and VS Code share a very similar UX concept

Despite these similarities, there are significant differences between Theia IDE and VS Code that are worth exploring. Let's now examine these differences in detail.

## Differences Between the Theia IDE vs. VS Code

### Features

Both Theia IDE and VS Code leverage the Monaco Code Editor and support LSP and DAP. Therefore, the core feature set of both tools is very similar. VS Code offers some exclusive features such as the 3-

way merge editor. In turn, the Theia IDE offers additional features such as detachable views (e.g., the terminal), an enhanced tab preview, and a customizable toolbar, enhancing the user experience. It also has features interesting for corporate environments such as the support for multiple extension registries (see next section).

Many relevant features for both tools are provided by VS Code extensions, which can be installed into both tools. Some of the latest feature additions in VS Code like Live Share, Copilot, and remote development were unfortunately provided as proprietary extensions. This means they are not open source, are exclusive to the VS Code product, and are not available in Theia, Code OSS, or VSCodium. In these cases, the Theia community often strives towards open alternatives. It already provides open-source remote container support, and an open alternative to Live Share. Native and transparent AI support is under development. We will go into detail about the support of VS Code extensions in the next section again.



In summary, the feature set of the Theia IDE and VS Code is very similar. The differences might be important for some users and influence their tool choice, for others the differences might not matter at all. We observe a tendency that some newer features in VS Code are closed source and sometimes subscription-based, a trend that might continue. In Theia, the available feature set is community-driven - contributors can influence it easily - and implementations are always completely open.

## Extensibility and Customization

VS Code supports extensibility primarily through VS Code extensions, which are limited to the APIs provided by VS Code. Theia IDE is compatible with the VS Code extension API and can host almost all VS Code extensions. As Theia has to implement any new API after VS Code has published it, it is usually one month behind. However, this does not really play a role in practice, as extensions usually do not pick up new API that fast. Extensions can be installed into VS Code from the VS Code marketplace, Theia uses openVSX by default (and can be adapted to any compatible registry). The VS Code marketplace hosts more extensions, but openVSX offers the important and popular ones as well (excluding proprietary ones mentioned before).

Additionally, Theia enables users to control which extension registry it connects to for installing VS Code extensions. While the Open VSX registry is a popular choice, users can also opt for other registries, including self-hosted ones. This control over extensions is particularly beneficial in corporate environments, where security and compliance are paramount, especially considering recent security risks associated with VS Code extensions.

While the VS Code extension model is simple and strives for isolation from the core application, it considerably restricts the extent of customization possible in VS Code. The Theia IDE is based on the Theia platform, which is by itself highly extensible and customizable, thanks to its modular architecture.

Via the underlying platform, Theia provides a second extension mechanism—called “Theia extension”—that is not limited to a specific API as VS Code extensions are. Therefore, adopters can modify the workbench, add custom features, and even remove default elements. Theia’s flexible architecture allows it to be adapted for a variety of use cases, from simple code editing to complex, domain-specific tools. This might not be relevant for users who just want to download and use a generic IDE. However, if you look for an IDE that you can customize to a specific use case or environment, e.g. a corporate set-up, this extensibility comes in handy. The flexible architecture also allows the Theia community to develop features that can then be optionally deployed into tools based on Theia. As an example, the Theia IDE, as a tool based on Theia, can decide whether it wants to include specific features provided by the platform and users can revise this decision.



### Most popular extensions for the Theia IDE from [Open VSX](https://open-vsx.org/)

In a nutshell, both VS Code and the Theia IDE support VS Code extensions and can therefore benefit from a huge ecosystem of available features. Additionally, the Theia IDE is based on the modular Theia platform and therefore allows deep customizations without forking the code.

## Open Source, Community and Governance

The Theia IDE is fully based on open source components and does not contain any proprietary parts. The Theia IDE (also the distribution) is provided under the commercially friendly Eclipse Public License. The Theia IDE benefits from a vendor-neutral, community-driven development and governance model under the Eclipse Foundation. This ensures a diverse range of contributions and influences, fostering innovation and reliability. The community-driven approach also means that users and organizations can have a direct impact on the project’s direction and the project is not controlled by a single stakeholder.

VS Code contains some proprietary components, which are not open source. The downloadable product of VS Code is provided under a proprietary Microsoft license (e.g. to allow telemetry and protect trademarks). Additionally, as mentioned before, some recent feature additions provided as VS Code extensions, such as LiveShare and Remote development support are closed source and also exclusive to VS Code. Some even claim that the best parts of VS Code are proprietary. At least there seems to be a trend for recent feature additions to be proprietary, as recently discussed for the AI integrations.

However, most of the code in VS Code is still open-source through Code OSS, which is primarily driven by Microsoft. This centralized control can streamline development but might not align with the needs of all stakeholders looking for a more collaborative and open governance model. In fact, it is sometimes hard for the community to influence the direction and add contributions to VS Code. As an example, a toolbar has been requested by many users, but the core maintainer decided against it. Also, exposing WebUSB to extensions was a long-running process in VS Code. The VS Code project is obviously strongly driven by the product requirements of VS Code, while Theia the platform is more open as it needs to fulfill the requirements of a variety of stakeholders.





## Adopters and contributors to Eclipse Theia

In summary, the foundation of VS Code and the Theia IDE are open source. Code OSS, the open source project behind VS Code is driven and controlled by Microsoft. Their strong contributions have created a great tool offering, as a user of VS Code you completely and strategically rely on Microsoft. Theia, the project behind the Theia IDE is developed in a vendor-neutral and diverse community governed by the Eclipse Foundation.

## Privacy and Telemetry

The Theia IDE places a strong emphasis on user privacy and has zero telemetry by default. There is an infrastructure available to turn on collecting usage data, which adopters can enable for their adoption use cases easily, but in Theia IDE no Telemetry Sender is implemented by default. VS Code collects telemetry data by default. As a user you can turn off several parts of it. However, there are reports (e.g. the article “VS Code - What’s the deal with the telemetry?”) claiming that you cannot really turn off telemetry in VS Code entirely at all. We will not dive into the details, as we did not research and confirm this ourselves. We recommend any users and adopters which care about privacy and to thoroughly research on this topic, though. This includes the tool itself, but also all used extensions, as they are generally capable of collecting and sending their own telemetry data.

## Deployment, Desktop vs. Cloud

VS Code is a desktop tool, with the cloud-based version (GitHub Codespaces) being a separate commercial offering that integrates with Microsoft’s cloud infrastructure. The server side of VS Code used in Github Codespaces has initially been closed source leading to several forks. In the meantime, there is a project called OpenVSCode Server providing a fork of the Code OSS code base to enable VS Code running in the cloud outside of Github Codespaces.

The Theia IDE is available as a desktop application and also provides a Docker container to be deployed in the cloud, providing seamless integration into various environments. Additionally, with Theia Cloud, there is an open source solution allowing orchestrating cloud deployments, i.e. if you want to provide a Theia-based application to multiple users, giving each of them their own workspaces.



In a nutshell, the Theia IDE and VS Code can both run on the desktop and in the cloud. For Theia the support comes fully open including an orchestration solution (Theia Cloud). For VS Code you can either use the commercial offering of CodeSpaces or rely on a fork of CodeOSS. The latter obviously forbids the use of proprietary extensions that are exclusive to VS Code.

## Conclusion

Both the Theia IDE and VS Code offer robust environments for development, each with distinct advantages. Theia IDE excels in extensibility, privacy, strategic independence and community-driven governance, making it a great option for developers seeking a customizable, open-source solution. On the other hand, VS Code provides a very well-known, extensible and popular editor. The Theia IDE has just recently been released. Therefore, the user base of VS Code is significantly larger and, as a result of the strong usage over the past years, VS Code underwent product hardening in more user scenarios.

At EclipseSource, we use both tools extensively, building solutions on both platforms, and we appreciate the unique strengths each brings to the table. Our goal with this comparison is to provide a neutral perspective, though achieving complete impartiality is challenging. Given that VS Code is much more well-known and has been available longer than the recently released Theia IDE, our comparison often emphasizes the Theia IDE's features and advantages. We assume readers are already familiar with VS Code's well-documented benefits.

We also cannot guarantee that every detail in this comparison is up-to-date, as some information is based on third-party research and aspects may change over time. If you find any inaccuracies or omissions, please contact us. We believe that every project and stakeholder should make an informed decision to select the ideal solution for their specific use case. The recent release of the Theia IDE enriches the ecosystem, providing users with a viable alternative to VS Code.

Ultimately, the choice between Theia IDE and VS Code depends on your specific needs and preferences. If you seek a powerful, feature-rich code editor with a vast extension marketplace and strong user base, VS Code is a compelling option. Conversely, if you focus on building custom tools, value flexibility, strategic independence and privacy, and wish to influence the development of your IDE, Theia IDE is an excellent choice.

If you are interested in building custom tools, IDEs or VS Extensions based on Eclipse Theia or VS Code, EclipseSource provides consulting and implementation services for [Eclipse Theia](#), for [VS Code](#) as well as

for [web-based tools in general](#). Finally, we provide [consulting and support for hosting web-based tools in the cloud](#). [Get in contact with us](#), to discuss your use case!