



# Visual Studio: Your Path to Modern Development

Protect your code, empower your developers

# Contents

|                                                                              |           |
|------------------------------------------------------------------------------|-----------|
| <b>Executive summary</b>                                                     | <b>3</b>  |
| <b>Why we recommend Visual Studio for Synergy DBL development</b>            | <b>4</b>  |
| <b>Synergy DBL Integration for Visual Studio</b>                             | <b>6</b>  |
| <b>Moving your DBL development to Visual Studio</b>                          | <b>7</b>  |
| Prerequisites and considerations                                             | 7         |
| Getting started                                                              | 8         |
| <b>Success story: Preparing for Continuity and Growth with Visual Studio</b> | <b>9</b>  |
| <b>Key takeaways</b>                                                         | <b>11</b> |

## Executive summary

Organizations with long-established, mission-critical applications face a common challenge: how to maintain and modernize their systems while ensuring operational continuity and preparing for future growth. By moving your development process to Microsoft Visual Studio, your company can preserve proven business logic while embracing industry-standard tools that enhance productivity, collaboration, and long-term agility.

This white paper outlines the benefits of Visual Studio, explains how to configure it for Synergy DBL development, and showcases real-world results from Synergex customer Sheet Metal Connectors, Inc. (SMC). Moving to Visual Studio was an important first step in modernizing their ERP system, enabling SMC to reduce risk, improve collaboration, and embrace current DevOps practices. The result was a more efficient, future-ready development environment that safeguards decades of investment while unlocking new opportunities for innovation.

# Why we recommend Visual Studio for Synergy DBL development

Developed by Microsoft, Visual Studio is a comprehensive integrated development environment (IDE) that supports a broad range of programming languages and development tools. As one of the world's most widely used IDEs, it's a popular choice for developers.

Synergy DBL integrated with Visual Studio provides an industry-leading and forward-looking development experience that enables teams to code more efficiently and integrate with modern DevOps workflows.

Here are some of the many reasons we recommend Visual Studio:

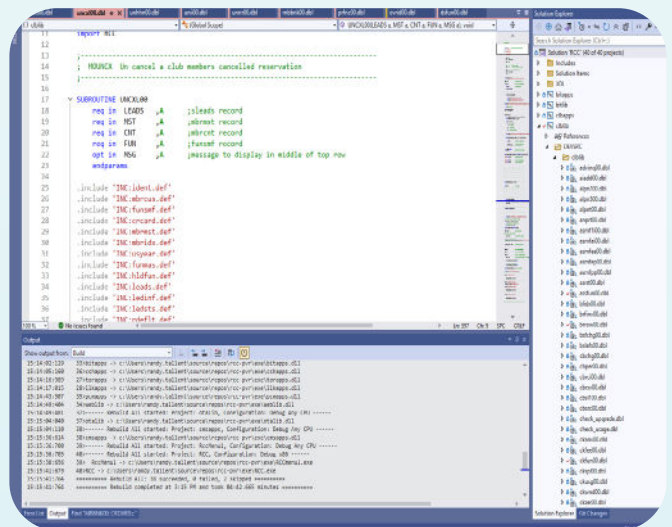
## ✓ Core tools that accelerate development

Visual Studio provides a developer experience that boosts productivity and lets your team focus on solving problems instead of performing repetitive tasks.

- IntelliSense offers real-time, context-aware code completion that helps you write code faster and significantly reduces the chance of typos and bugs.
- Auto-formatting automatically aligns and organizes code according to best practices that you set, improving code readability and ensuring that your team adheres to consistent coding standards.
- Go to Definition and Go to References let you instantly navigate to any subroutine, variable, or symbol in your codebase, without leaving the IDE or hunting for files.
- Visual Studio's advanced debugging environment allows you to set breakpoints, watch variables, inspect a call stack, and step through code in a user-friendly interface.

## ✓ Customizable and extensible environment for ultimate flexibility

Visual Studio allows you to customize your IDE to suit specific project needs and workflows and provides powerful tools for refactoring, code cleanup, navigation, dependency management, and more. Such features help ensure your code remains organized, efficient, and easy to update and make it easier to manage your development process.



## ✓ Version control and collaboration, all in one place

Integrated Git support with GitHub, Azure DevOps, or any other Git-based service makes version control seamless. You can clone repositories, create branches, stage changes, commit, and push updates without leaving the IDE. Built-in tools for reviewing pull requests, resolving merge conflicts, and viewing commit history make it easier for teams to collaborate and stay in sync, whether they're in the same office or working remotely.

## ✓ Built for modern build and deployment pipelines

Visual Studio's build engine, MSBuild, gives developers precise control over how applications are compiled, packaged, and deployed through common properties and environment variables. It also integrates with modern CI/CD pipelines, enabling automated testing and continuous integration, and its cross-platform support means you can target cloud, on-prem, or hybrid environments.

## ✓ Integrated testing for more reliable software

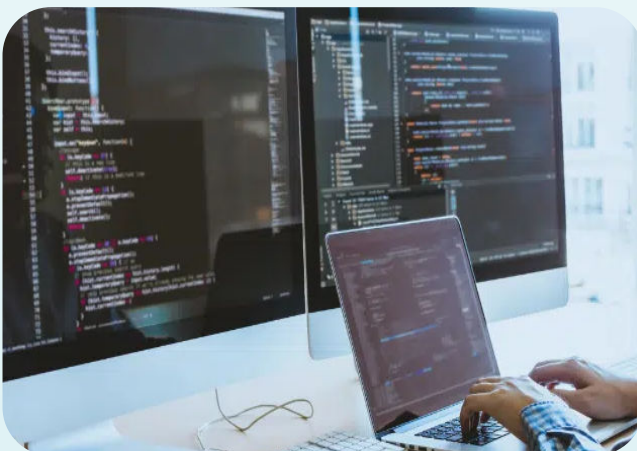
Built-in support for popular testing frameworks enables developers to create, run, and manage tests inside the IDE, helping you catch errors early and deliver reliable software while reducing time spent on troubleshooting and manual testing.

## ✓ Easier developer recruitment and onboarding

Because many developers already have experience with the environment, using Visual Studio simplifies recruiting and streamlines onboarding, allowing new team members to become productive more quickly.

## ✓ Continuous evolution to keep your software relevant

Microsoft's ongoing investment and active engagement with the developer community ensures Visual Studio is always improving and evolving in step with user needs, delivering state-of-the-art features, performance enhancements, and timely fixes.



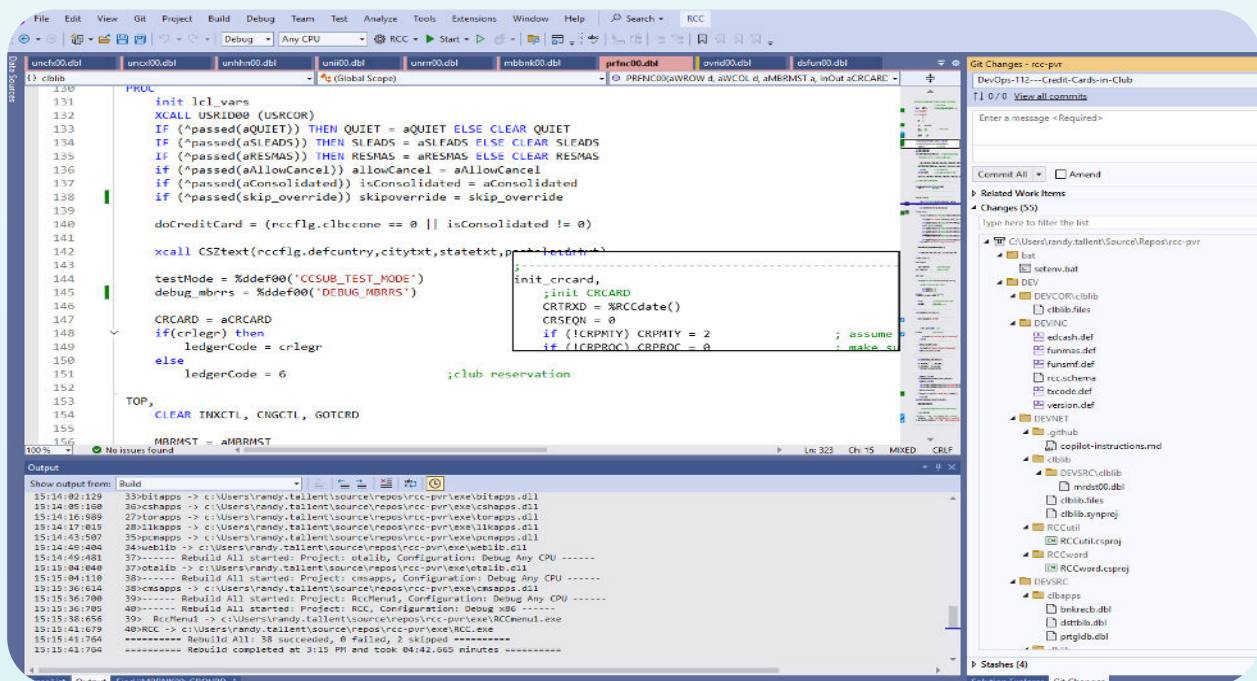
# Synergy DBL Integration for Visual Studio

Synergy DBL Integration for Visual Studio (SDI) is a development environment extension that brings Synergy DBL programming directly into Microsoft Visual Studio.

SDI delivers

- Project system integration to allow you to use Visual Studio's solution/project model to organize and build Synergy applications
- Custom build targets that cater to Synergy project types, including ELBs, OLBs, scripts, and Repository
- The ability to build Synergy .NET components or hybrid applications for Windows or Linux that combine Synergy DBL with C#, VB.NET, and other .NET technologies
- Support for multi-mainline projects
- Version targeting, enabling you to develop using the latest tools while targeting older Synergy versions
- Code conversion tools that assist in migrating traditional Synergy code to .NET
- A custom Synergy terminal with project-aware environment variables for smoother workflows
- Support for debugging traditional Synergy programs in Visual Studio, including remote debugging on Unix and OpenVMS

Synergex's continued advancement of SDI underscores its commitment to providing DBL developers with modern tools and advanced features. By staying current with Microsoft releases, Synergex ensures developers have access to the latest capabilities quickly and without disruption.



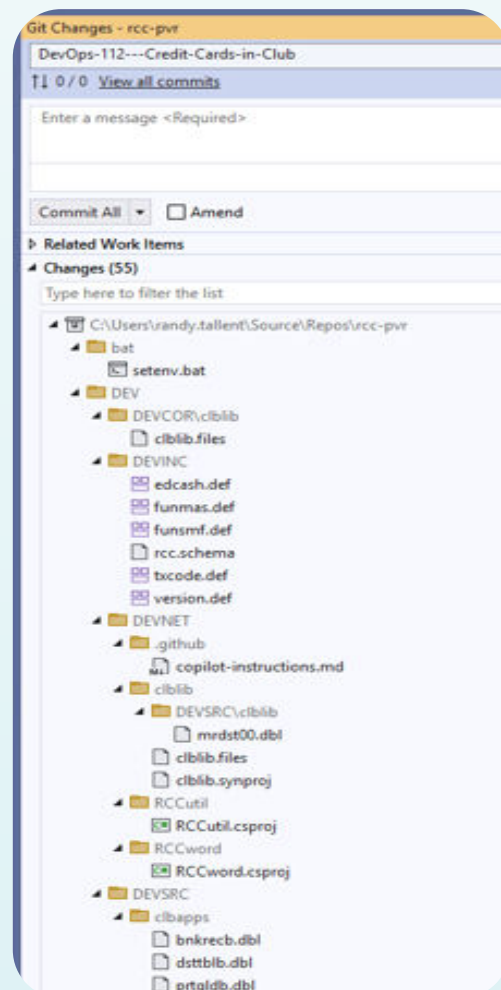
# Moving your DBL development to Visual Studio

Moving to Visual Studio might be easier than you expect. Older source code is often the easiest to migrate, since it tends to have fewer dependencies and less architectural complexity. Depending on the state of their source, Synergy developers often achieve a working Visual Studio solution in as little as a few weeks.

## Prerequisites and considerations

Before setting up Visual Studio for Synergy DBL development, there are a few prerequisites to meet and decisions to make that will shape your configuration:

- Synergy/DE (preferably the most current version) must be installed on Windows.
- Determine which Visual Studio product you need: Visual Studio Community (free, for nonprofits or very small organizations), Visual Studio Professional (for commercial teams or small-to-medium businesses), or Visual Studio Enterprise (for large enterprise organizations). All use the same core IDE, but Enterprise is significantly more expensive and includes specialized tools.
- Decide what you want to do with your source files and source tree: keep them as is (and reference the files in their current locations) or move/copy the files into Visual Studio project locations. By default, when you create a new Synergy project and start adding source files, it'll create copies of those files in the project's source directory while preserving the original directory structure. Leaving the source files in their current locations makes it easier to stay synchronized with code maintained on other operating systems or edited with other tools.
- Decide what source control you'll use. Visual Studio has loads of plug-ins. If you already use a product like SVN, you can get the SVN plug-in to Visual Studio and continue using it. If you don't already have source control, we recommend using a flavor of Git, either from GitHub, a cloud provider, or Azure DevOps.



# Getting started

The following are the steps Synergex's Professional Services Group (PSG) typically takes when helping customers adopt Visual Studio.

## 1. Install Visual Studio and Synergy DBL Integration.

- Download and install the latest version of Visual Studio from the official [Microsoft Visual Studio website](#).
- Download Synergy DBL Integration for Visual Studio (SDI) from [Synergex's Downloads page](#) and follow the installation instructions.

## 2. Configure your development environment.

- Open Visual Studio and configure the Synergy DBL environment settings for your project requirements.
- Add any environment variables you need, along with include directories, common properties, script files, and other artifacts.
- Configure project properties such as target platform and output directories. Use relative paths.

## 3. Create a new Synergy DBL project.

- In Visual Studio, create a new project and select the Synergy DBL project template.
- We recommend starting with an ELB/OLB project, then adding five or ten source files at a time rather than the whole directory.

## 4. Compile as you go and iterate on your code.

- If it doesn't compile, solve any errors and compile again. Errors are expected and are a good thing!

## 5. Continue building out your solution by adding other projects (script, repository, multi-mainline, etc.) as needed.

## 6. Integrate with Git (or the source control system you selected) to manage code versions and collaborate with team members.

You can find more detailed information about getting started with Visual Studio and developing Synergy applications in Visual Studio in the [Visual Studio](#) and [Synergy/DE documentation](#).





## SUCCESS STORY

# Preparing for Continuity and Growth with Visual Studio

Sheet Metal Connectors, Inc. (SMC), a leading manufacturer in the HVAC industry, successfully moved their development process to Visual Studio to protect business continuity and improve developer experience and productivity.

## Background

SMC's Synergy-based ERP application is a critical component of their operations, encompassing modules like Accounts Payable, Accounts Receivable, General Ledger, and Order Processing. Originally based on industry-specific software, the application had been heavily customized and maintained for over 30 years.

## Challenges

Outdated tools and reliance on a single key developer posed operational risks. They needed to adopt industry-standard development tools to facilitate collaboration and ensure business continuity.

## Solution

SMC moved their development process to Visual Studio, leveraging the IDE's advanced features to modernize their workflow. The project enabled SMC to adopt modern DevOps practices, thereby improving collaboration and operational efficiency.

## Results

- Enhanced development capabilities through Visual Studio's comprehensive toolset.
- Reduced operational risks by transitioning to a modern, industry-standard development environment.
- Improved collaboration among developers, leading to more efficient maintenance and enhancement of the ERP system.

SMC's story exemplifies the benefits of developing with Visual Studio, demonstrating how modernization can lead to improved development efficiency and business operations. For more information, read the full case study.



Learn more about SMC's journey

[Read the Case Study](#)

# Key takeaways

Visual Studio's comprehensive toolset gives development teams a contemporary, flexible environment that grows with their needs. For businesses running long-established applications, it offers a strategic path to modernization that preserves existing investments while reducing risk and unlocking new opportunities for growth.

In short, Visual Studio empowers businesses to

- Protect and extend the value of their existing applications
- Adopt modern development tools without sacrificing stability
- Improve collaboration and productivity among developers and across teams
- Position themselves for scalability and long-term success

Together, Visual Studio and SDI provide the foundation for a development environment that meets today's demands while positioning your team for whatever comes next.

## Want to learn more?

For more information, contact Synergex at +1.916.635.7300 or [synergy@synergex.com](mailto:synergy@synergex.com).



## Get a boost from Synergex Professional Services Group

Synergex Professional Services Group can assist you with your move to Visual Studio.

[Learn About PSG](#)



Synergex provides software development tools, application integration technologies, and expert consulting services to help enterprise application developers retain their software investment, keep up with advancing technologies, and bring their applications into the future. For over 50 years, Synergex technologies have been the foundation of applications that drive commerce around the world. Every day, millions of users interact with these systems in e-commerce, global logistics, manufacturing, healthcare, and other industries.