

## Razor for Agile CM Practices

Razor is a an integrated software tool that helps manage the complexity of your software development by improving software quality and reducing risks through agile practices.

Visible Systems Corporation

razor\_sales@visiblesystemscorp.com

## Razor for Agile CM Practices

### Razor for Agile

Stories are organized according to their status (Backlog, Assigned, In Process, Completed)

Stories can be viewed or updated at any time by an authorized engineer.

Stories are managed in their own swimlane (indicated by their state).

Stories are individually described with due date setting.

Sprints are identified by name, state, end date.

Stories and Sprints are tightly integrated within with Issues, Versions, and Baselines within Razor.

### Razor Provides

- Structure
- Control
- And Repeatability
- For Your Process

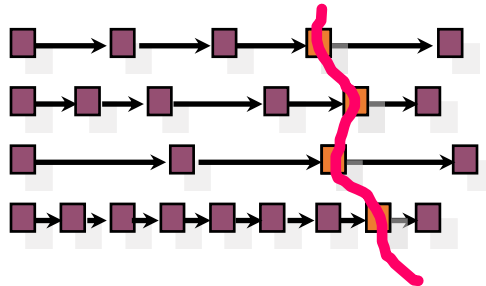
### Use Cases

- Issue, Defect, and Problem Tracking
- Defect Management
- Version Control
- Change Management
- Document Control
- Software Configuration Management
- Software Process Management
- Software Quality Assurance
- Release Management

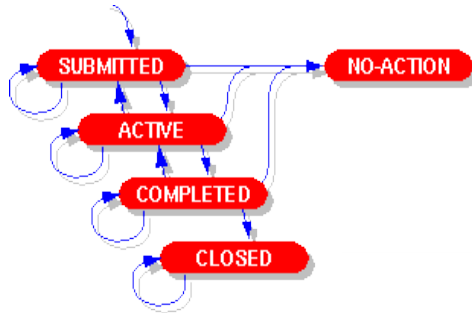
### Why Use Razor?

- ❖ It's Easy to Use
- ❖ Configurable
- ❖ Affordable
- ❖ Increases productivity
- ❖ Improves product quality
- ❖ Enhances teamwork
- ❖ Saves time and money

*Parallel Development*



*Issue, Defect, and Problem Tracking*



Razor can be used out of the box by downloading it from [visiblesystemscorp.com](http://visiblesystemscorp.com). Razor can be customized to handle any development process for your product or projects. It easily scales across the full lifecycle of a project or across the plant or enterprise. Razor is optimized for Software Development, includes Agile features, is well suited for ISO 9000, CMM, is methodology independent, and supports any SQA initiative, or any e-business or web development project.

- EASY TO LEARN, USE, AND MANAGE
- HIGHLY CUSTOMIZABLE and can tailor to your internal development process
- Manages Change throughout the software development lifecycle
- SIMPLIFIES generation of release notes and development support metrics
- Shortens the systems development life cycle with the continuous delivery of high quality software

Razor provides full accountability and enforcement of your development process. Triggers may be attached to nearly every Razor action. All user activity is logged for audit traceability. You can customize your state transition model to your needs. Some customers have 10-15 unique steps in the resolution of an issue; others have 4-5 steps. It's up to you. **Razor conforms to YOUR process**, not the other way around.

Razor can instantly recreate any release ever created. One manufacturing customer was just blown away by this capability. They couldn't believe that Razor could go all the way back to a release created in 1996 (or recreate any release that had ever been created on the production floor).

Razor can be configured to automatically extract and build new versions of your software every day.

**Razor can be configured to your internal needs to add Structure, Control, and Repeatability to your development process and can be scaled for projects of all sizes.**

## Who's Used Razor?

### AEROSPACE

Aero Simulation Inc., United Launch Alliance, Boeing, Lockheed Martin, Kratos, Orbital Sciences

### ELECTRONICS AND SEMICONDUCTORS

Vitesco Technologies, Continental

### DEFENSE CONTRACTOR

General Dynamics, Integral Systems, E-Systems, SAIC, L3Harris, Northrop Grumman

### GOVERNMENT

Social Security, Air force Office of Scientific Research (AFOSR), NASA, NOAA, National Buoy Data Center

### CLOUD TECHNOLOGY

Software One

### MINING

Alliance Resource Partners

### PHARMECUTICALS

Roche

### SCIENCE AND RESEARCH

Los Alamos National Labs, Sandia National Labs

## For Problem Tracking

Through a fully customizable interface, **YOU** define the type of information you want tracked, and how you want it displayed.

Razor provides checkboxes, textfields, X-of-many selectors, lists, timestamp fields.

It can filter and sort the display through any combination of fields.

You can define you own rules for signature chains and email notifications.

You have at your fingertips, a powerful email interface for entry or querying information.

You can implemnet your process through the use of triggers.

It has an intuitive graphical interface.

## For Version Control & Change Management

Razor easily handles any type of file. It allows users to check files in and out of the system for edits, while automatically tracking the who, what, where, and why of their efforts.

- ❖ Simple drag-n-drop integration with the problem tracking system gives full bi-directional accountability.
- ❖ Quickly review the change history of any file in the system.
- ❖ Graphical merge to support parallel development.
- ❖ Full Access Control.

## For Release Management

Razor lets you generate named files etc., to define specific product builds.

This can be done either in a file-by-file basis, or through broad statements such as "include all C files as the version that passed release testing" or "update my definition based on changes related to a list of issues."

These are version-controlled, living file sets, which evolve along with your projects.

- ❖ Complete control and automation over the build process.
- ❖ Compare any two releases.
- ❖ Easily and fully recreate any build known to the system.

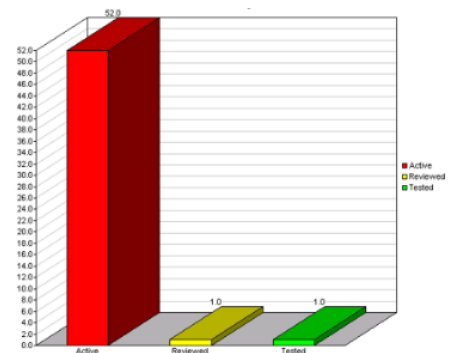
## Razor's History

Razor was born as a part of the Seawolf-class nuclear-powered submarine program to develop to the BSY-2 system used for sonar tracking, monitoring, and launch of onboard weapons. This large software development effort integrated the work of **300 software engineers**.

NASA used Razor for the duration of the Space Shuttle program for its Checkout and Launch Control System to design consoles to communicate with ground support for measuring 40,000 temperatures, measurements, flow rates, turbine speeds, voltages, valves, switches, etc. This program involved efforts of more than **200 engineers**. Razor was used for more nearly fifteen years until the program ended.

Razor can easily be scaled up and customized to meet any software organization's needs. To give you an idea of what Razor can accomplish. A large Texas automotive chip manufacturer who's been using Razor for more than twenty-five (25) years for embedded software development. During that tome, Razor has been managing the software development for **seventy software engineers** for a billion-dollar automotive chip line – with no major support issues.

Whether you are managing the configuration of a single machine or thousands, Razor can manage them with precision – across the plant, across the country, or around the world.



*Generates Metrics & Reports*

## Razor for Agile CM Practices

- ❖ Sprints
- ❖ Issue and Bug Tracking
- ❖ Stories
- ❖ Versions
- ❖ Baselines

Selected Sprint

Sprint Name: Test Sprint 1      Sprint Start Date: 3/15/2022 2:43:27 PM      Sprint End Date: 4/1/2022 2:43:27 PM

Backlog	Story	Assigned	In Process	Completed
	Issue: Rad...3 This is a story in the submitted state			
		Issue: Rad...4 This is a story in the assigned state Assigned to: John Doe Due Date: Mar 30, 2022		
			Issue: Rad...5 This is a story in the In Process State Assigned to: John Doe Due Date: Mar 31, 2022	

Refresh  
New Story  
Close

**Brief Description of this Story**

Create a new Story here

Minimum Due Date  
Thursday, March 31, 2022

**Detailed Description**

This is a demo of the creation of a new Story. The Story is introduced into the Issues tool where it can be used for Versions file manipulation and Baseline creation.

Cancel  
Submit

*For Sprints, when a Story is moved to the completed state, a Snapshot in Baselines is automatically created.*

Versions Universe: MoreRazorTests Group: RazorTest

File Edit View Utilities Scripts Commands Groups Help

include	
acceptor_socket.h	1.2
accept_issue.cursor	1.1
access_list.h	1.1
action_funcs.h	1.1
admin_attr_defs.h	1.1
admin_attr_funcs.h	1.1
admin_tool.h	1.1
approval_strings.h	1.1
attr_funcs.h	1.1
blowfish.h	1.1
browse_glyph	1.1
browse_file.h	1.1
checksum.h	1.1
chooser_glyph	1.1
chooser_funcs.h	1.1
client_server.h	1.1
comm_funcs.h	1.1
convert_mode.h	1.1
crypt.h	1.1
date_funcs.h	1.1
debug.h	1.1
deleted_info.h	1.1
dir_funcs.h	1.1
dir_tree.h	1.1

Change Group: RazorTest

Version Functions

Check Out for Edit  
Check Out Read Only  
Check In  
UnCheck out  
Branch  
Terminate Branch  
Introduce

Exit

## Razor 6 in an Enterprise Environment

In an Enterprise configuration, each site will contain its own server and database.

The configuration for each site should follow Visible Systems site configuration guide.

Distributed Databases and Servers reduces loads and data contained on the site.

Each database must contain the information for how to contact all sites. The information will contain the fully qualified name of the server and all universes present at that site.

Note: Each site can contain restrictions on which Servers/Universes are available.

The developer will select the site and then the Universe that is located at that site

All transactions such as new Issues, work in Versions or Baselines will be conducted at the remote site.

Scripting for all tools will be executed at the remote site level.

When using Versions or Baselines files can be Introduced, Edited or extracted at the local level.

In an Enterprise Environment, we recommend that MySQL replication be considered.

Also, using a commercial version of MariaDB is recommended (Percona.com or MariaDB.com).

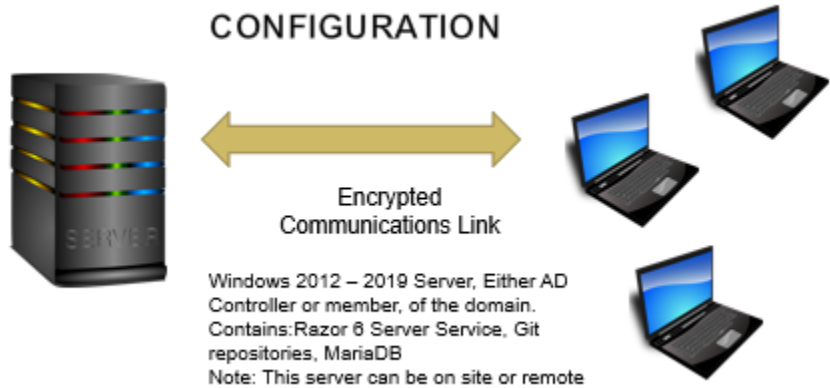
With the commercial version of MariaDB.com, a front end called Maxscale is included.

Maxscale will distribute the load across the replicated servers.

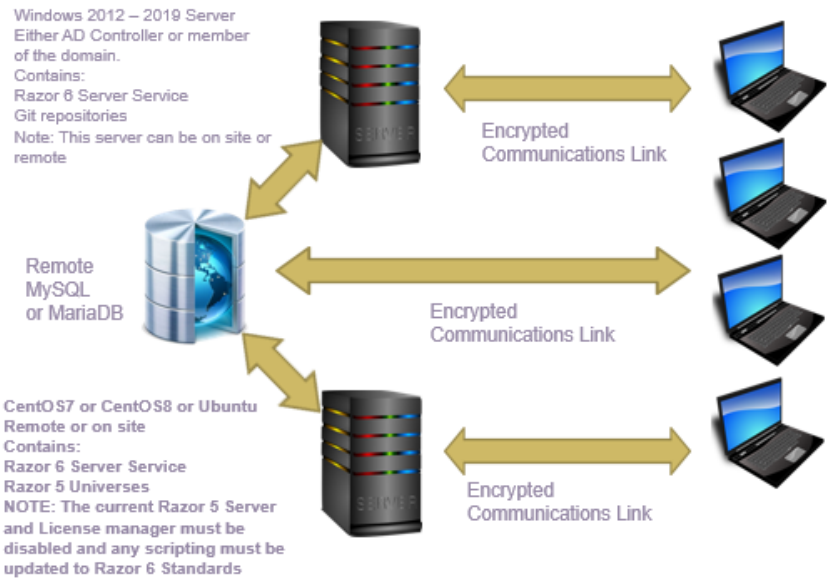
Cluster Control from SeveralNines.com is an excellent monitoring and control tool for a replicated cluster.

For more details, contact Visible Systems for information on a replicated server setup for your Enterprise solution.

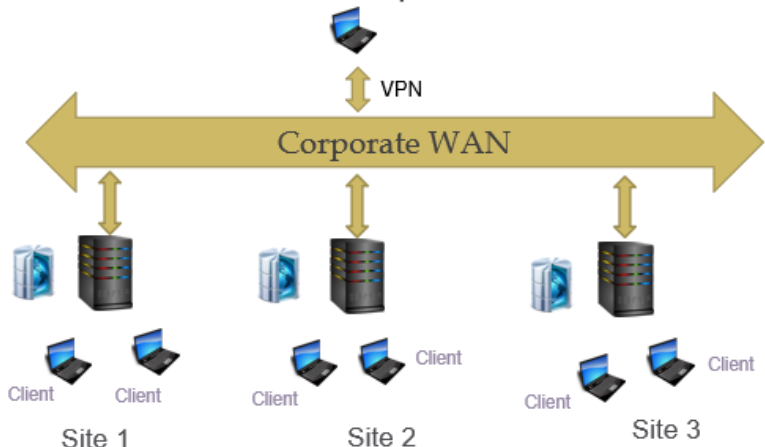
### SIMPLE WINDOWS BASED RAZOR 6 CONFIGURATION



### Complex Razor 6 and Razor 5 Configuration



### Razor 6 in an Enterprise Environment



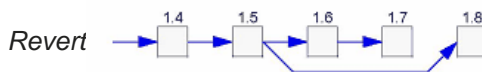
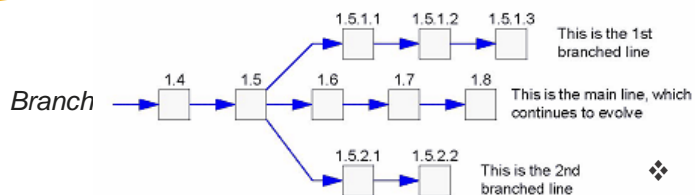
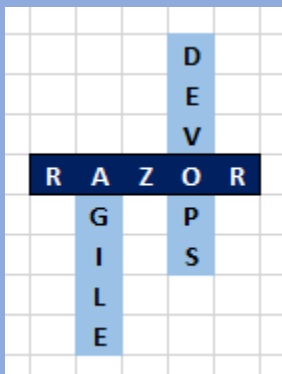


## Razor Uses Git

Razor uses a relational database for its metadata and a Git repository for versioning files. The versioning of files is contained in the MariaDB database because Git doesn't include version tracing. The roadmap for what versioned items means is stored by Razor in the MariaDB. Razor understands every revision of a file, or new iteration of a file and knows how to put it all together. Razor brings to the table an organized process with business rules behind it, allowing developers to understand when an issue state gets promoted and gets stored in the MariaDB.

DRAG AND DROP Your Git files into Razor to start your first Razor project.

Razor includes a Powerful Command Line Interface, Supports any Scripting Language, and is highly Secure. All Communications



- ❖ Problem / Bug Tracking
- ❖ File Version Control
- ❖ Change Management
- ❖ Release Management

**The Razor Trilogy:** issue and defect tracking, revision/version control, and release management all tightly meshed together throughout the development lifecycle in one integrated toolset, rather than a conglomeration of independent tools.

## Supported Platforms

RAZOR'S COMPONENTS:

- ❖ ADMINISTRATION,
- ❖ SERVER,
- ❖ CLIENTS

### WINDOWS

RUNS ON WINDOWS 10, 2016, 2019 (64BIT)  
 USES .NET 4.X FRAMEWORK  
 RUNS IN A VMWARE VIRTUAL MACHINE  
**MARIADB** IS USED TO STORE (SETTINGS, CONFIGURATION, USER INFORMATION, ENCRYPTION PROFILE, LICENSE, ETC.)  
 OPEN SOURCE AND SPAWNED FROM MYSQL  
**GIT** IS USED TO STORE REVISIONS OF ISSUES, FILES, AND BASELINES  
 CONTAINS THE GIT FILE AND BASELINE HISTORY

### LINUX

OPEN-SOURCE FILE REVISION ENGINE  
 AVAILABLE ON LINUX  
 LINUX RUNS ON UBUNTU 20, RHEL 7/8, CENTOS 7/8 (E.G., 3.X, 4.X, OR 5.X 64BIT KERNELS) USING MONO  
 RUNS IN A VMWARE VIRTUAL MACHINE

### SERVERS

WINDOWS 2012 – 2019 (ON-SITE OR REMOTE)

\* Razor runs simultaneously in a mixed Windows and Linux environment.

## Evaluation and Demonstration

A FREE evaluation license of Razor is available for all supported platforms that can be downloaded from our website at: [visiblesystemscorp.com](http://visiblesystemscorp.com).

### Licensing

Razor uses a concurrent licensing approach. Pricing and additional product information is available at:

[visiblesystemscorp.com](http://visiblesystemscorp.com)

Razor is available as an on-premise or subscription license.

### Service and Support

Service and Support

We are dedicated to customer satisfaction, which has demonstrated by our top-notch support department that has supported Razor for more than 27 years.

Razor Consulting/Training Services

Razor consulting services are available upon request for planning, migration, installation, and customization needs.

Training classes are also available.

For more information on licensing, training classes, consulting, or a price quote.

Email: [razor\\_sales@visiblesystemscorp.com](mailto:razor_sales@visiblesystemscorp.com)

Or, Call: (781) 367-9509

