

Cummins Vehicle Mission Simulation Tool: Software Architecture and Applications

Nagesh Belludi, Jeremy Raymond, and Joshua Receveur
Cummins Inc.

Arvind Hosagrahara
The MathWorks, Inc.

Copyright © 2010 SAE International

ABSTRACT

This paper presents the business purpose, software architecture, technology integration, and applications of the Cummins Vehicle Mission Simulation (VMS) software. VMS is the value-based analysis tool used by the marketing, sales, and product engineering functions to simulate vehicle missions quickly and to gauge, communicate, and improve the value proposition of Cummins engines to customers.

VMS leverages the best of software architecture practices and proven technologies available today. It consists of a close integration of MATLAB and Simulink with Java, XML, and JDBC technologies. This Windows compatible application software uses stand-alone mathematical models compiled using Real Time Workshop. A built-in MySQL database contains product data for engines, driveline components, vehicles, and topographic routes. This paper outlines the database governance model that facilitates effective management, control, and distribution of engine and vehicle data across the enterprise.

This paper also presents four case studies on the applications of VMS to augment Cummins engineering and sales competencies by helping customers understand vehicle performance, optimize vehicle configurations, improve fuel economy, and reduce greenhouse gas emissions.

INTRODUCTION

The commercial vehicle industry has become increasingly aware of the interconnectedness of the components of a vehicle. Companies now research and engineer products around how each component interacts with other elements of the vehicle. Commercial software tools such as AVL Cruise, AVL ISAC, Advisor, GTDrive, and PSAT have evolved to support holistic vehicle integration, evaluate energy efficiency, and analyze component-level and system-level performance.

To supplement the above commercial software tools, engine and vehicle manufacturers have also developed in-house performance modeling software customized around their operating needs. Cummins uses an array of engine and vehicle validation tools as illustrated in Figure 1. Our suite of simulation tools, collectively called “CyberApps”, has been developed and validated over a span of 14 years. It operates a Unix based modeling platform and offers increasing degrees of fidelity by substituting mathematical models with hardware-in-loop

units of engine and vehicle subsystems. The simplest of these tools, “PureSim”, operates exclusively on MATLAB and Simulink models. It is this version of CyberApps that we have ported to Windows and adapted as the mathematical backbone of the analysis software that we describe in this paper. A discussion of the CyberApps mathematical models and validation process is outside the scope of this paper.

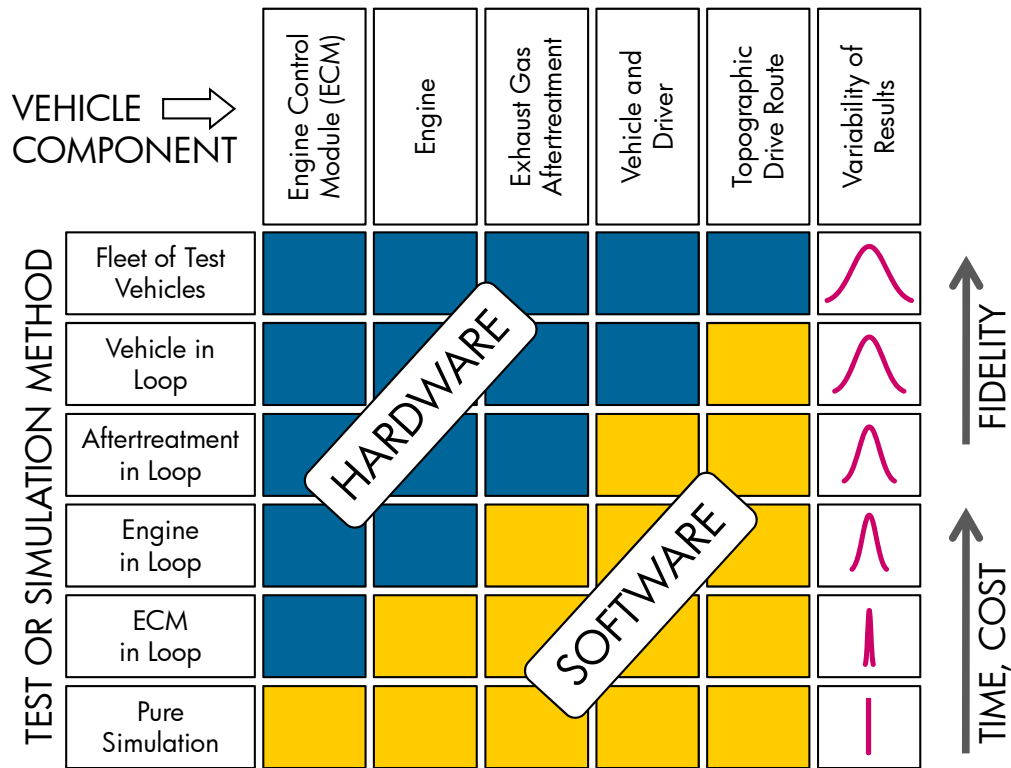


Figure 1: Vehicle and engine testing and simulation tools used at Cummins

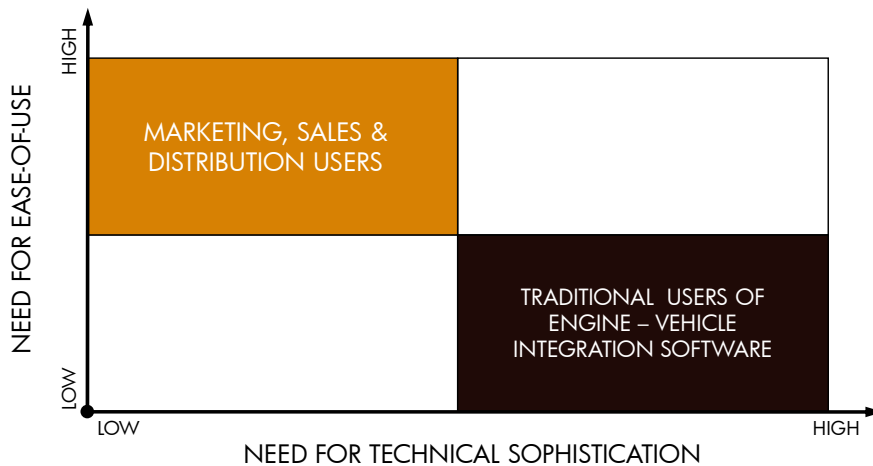


Figure 2: Segmentation of users of engine-vehicle integration analysis software

The in-house and commercial analysis tools we discussed above cater principally to the needs of advanced product engineering and research groups as illustrated in Figure 2. Consequently, these tools are complex, difficult to access, time-consuming and expensive for users outside of the engineering functions. These tools

typically require that the user have a firm grasp of the scientific knowledge of engines, drivelines and vehicles to be able to identify, quantify and model vehicle missions and subsequently interpret the technical results. Given the comprehensiveness of these analyses tools, simulations often run in real-time or are time-consuming. Therefore, the in-house and commercial analysis tools are too sophisticated and inconvenient for everyday use by marketing and sales functions to support the immediate needs of customers and simulate their vehicle missions during point-of-sales interactions with customers.

A VALUE-BASED ENGINEERING TOOL FOR MARKETING AND SALES

At the heart of business marketing is the fundamental need to understand the value of a company’s product to its customers. Given the opportunities in today’s commercial vehicle marketplace, the ability to gauge and communicate the value proposition of a product has never been more critical. In particular, at Cummins, the ability to demonstrate persuasively the value of our engines to customers hinges on our ability to model customer-specific missions and estimate the fuel and carbon efficiencies of our engines. We identified the need for a handy software tool that can enable our worldwide sales force and distribution network to analyze what-if scenarios for our customers, select the vehicle configurations that are most suitable for their missions, and model fuel consumption. This software can equip our salesforce with the right technical knowledge to contribute meaningfully to the buyer’s decision process at different points during the sales interactions.

To meet the unique requirements for a value-based engineering tool for the Cummins sales force, we developed the proprietary Vehicle Mission Simulation (VMS) software. From a product placement perspective, VMS is a “disruptive technology” – we ported PureSim, the Unix based CyberApps simulation tool hitherto available only to advanced users in the product engineering and research and development groups, to the Windows platform and delivered a simple-yet-sophisticated-enough engineering tool for marketing and sales functions. VMS has a small footprint, is easy to use, and is downloadable free of charge by any user within Cummins.

From a software architecture perspective, VMS leverages an integration of multiple technologies for efficient implementation, deployment and administration, as we will explain further in this paper. We bundled the engine-vehicle analysis models in Simulink with a secure MySQL database of engine and vehicle parameters and compiled a stand-alone graphical user interface that delivers fast non-real-time simulation results in the form of graphs, reports, and detailed logs.

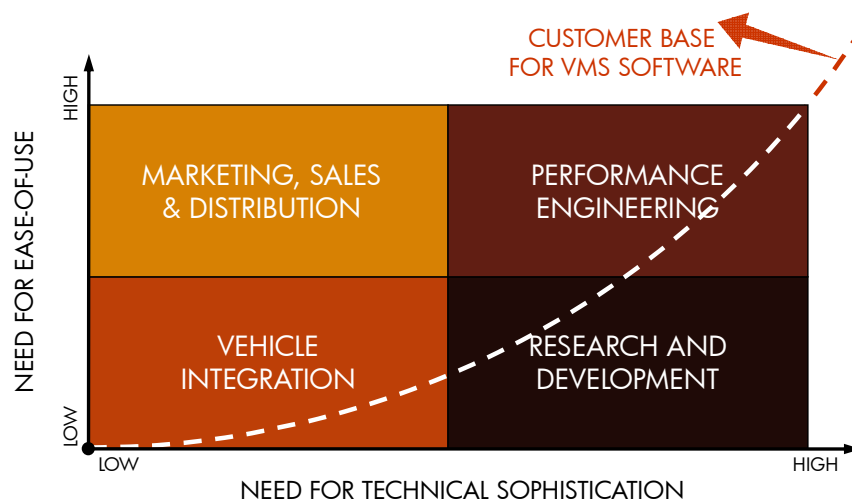


Figure 3: VMS is comprehensive enough to appeal to a wide realm of users within Cummins

We originally built VMS around the requirements of the marketing and sales functions. Nevertheless, we quickly scaled up its technical capabilities and made it comprehensive enough to appeal to a wide realm of users within Cummins as illustrated in Figure 3. Vehicle integration engineers use VMS to profile various applications in terms of operating duty cycle. They also use VMS to study gearing recommendations for better fuel economy without sacrificing vehicle gradeability (defined as the maximum road grade a vehicle can climb at a given speed,) startability (the maximum grade on which a vehicle can be put into motion,) and driver satisfaction as measured by frequency of gear shifting. The map-based modeling of engine performance in VMS helps performance development engineers get rapid feedback on successive iterations of engine calibrations and torque curves, and optimize the energy efficiency of Cummins engines in vehicle dependant missions.

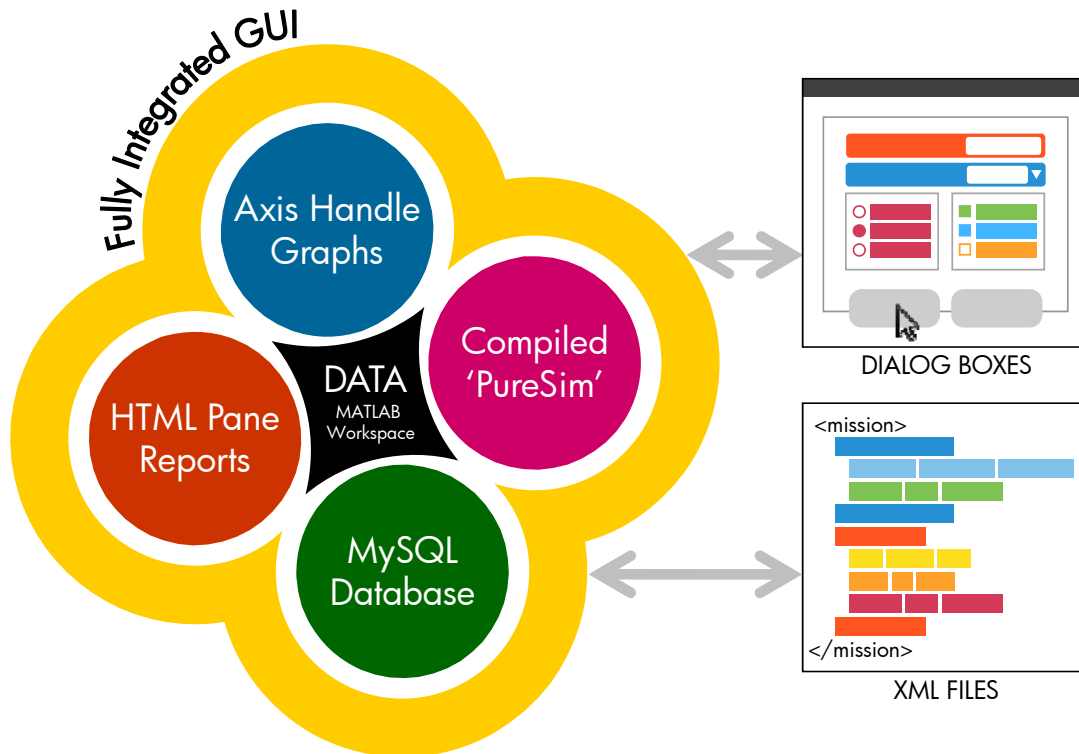


Figure 4: Overview of the VMS software architecture

VMS SOFTWARE ARCHITECTURE

VMS application encompasses a modular design that leverages existing technologies of MATLAB and Simulink. In this section, we describe the significant building blocks of VMS as illustrated in Figure 4, and present a behind-the-scenes look at how we put all these elements together into a stand-alone MATLAB application.

As we discussed in a preceding section, we adopted CyberApps PureSim, an internally developed engine-vehicle simulation tool, as the backbone of VMS. The fact that PureSim is based in MATLAB and Simulink, coupled with the computational power, speed, and ease of use of MATLAB, was the primary driver in our selection of MATLAB as the development environment for VMS software. Furthermore, the ability to develop and distribute royalty-free applications that run in self-sufficient MATLAB sessions on users' computers underscored our selection of the MATLAB graphical user interface development environment (GUIDE) for

software development. GUIDE also facilitates the seamless integration of JAVA Swing elements and content-rich features we have described further in this section. For the VMS database, we could choose any JDBC-compliant relational database management system (RDBMS) since MATLAB's database toolbox provides a single, open, plug and play interface to any RDBMS. We chose the MySQL Community Edition because of its ease of installation on users' computers, small install footprint, and ease of database administration.

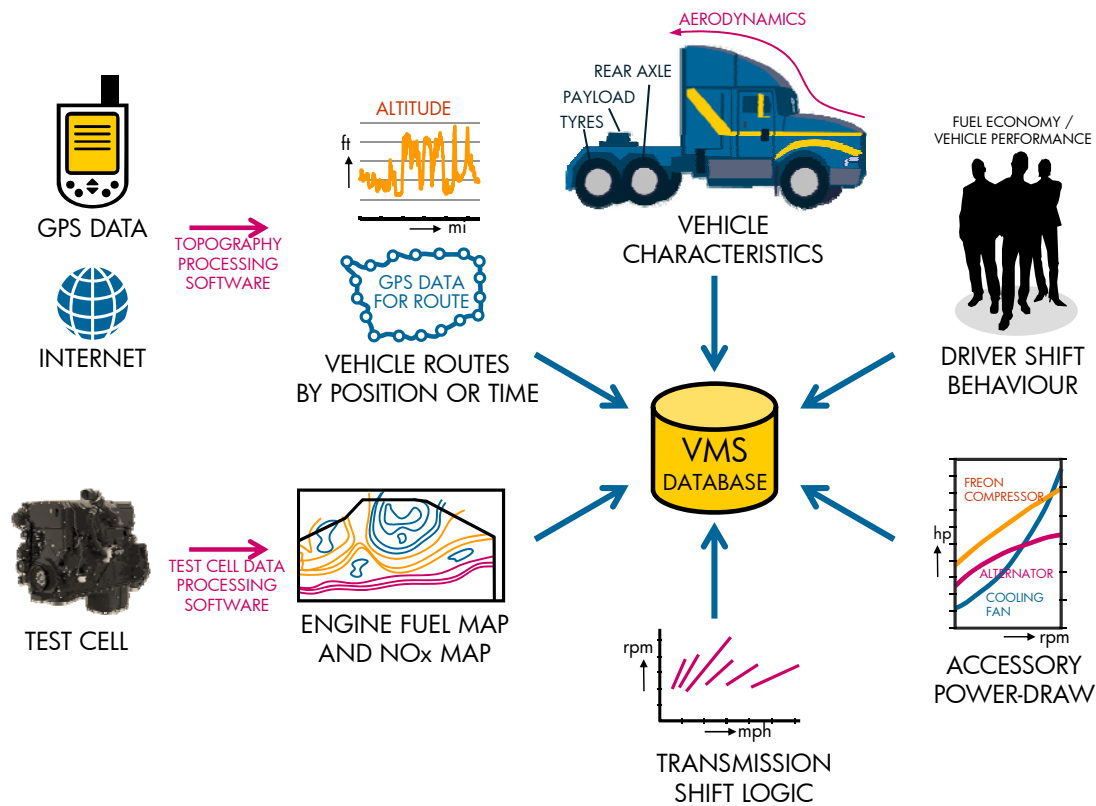


Figure 5: Elements of the inbuilt-MySQL database in VMS

ENGINE AND VEHICLE DATABASE

VMS includes a MySQL database as the relational database management system for engine fuel maps, parameters of vehicle and driveline components, topographic information of vehicle routes and all other data used by the PureSim analysis modules, as illustrated in Figure 5. The database contains legacy and contemporary information that Cummins has accumulated from more than two decades of active partnership with vehicle OEMs, and transmission, axle, and tire manufacturers.

We have established a decentralized infrastructure for the administration of the VMS database. As illustrated in Figure 6, principal to this infrastructure is our VMS team that serves as the global administrator. Our team designs the database architecture and relationship structures based on the needs of the back-end computational models in PureSim. To assist us in maintaining, overseeing, and distributing the VMS database, we have appointed application engineers around the globe as local database administrators. These local administrators serve as stewards of engine, driveline and vehicle data to support the operating needs of VMS users in their specific geographical or functional areas. They ensure data availability and timely distribution of their VMS databases to appropriate users in their domains.

The VMS local administrators are also responsible for custodial compilation of local data. For example, the local database administrator in India populates and supplies that geography's database to local VMS users. He works with the product development teams in India to collect, process and maintain fuel maps of locally-developed engines. In his primary role as a Cummins application engineer in India, he has previously established links in the local commercial vehicle industry and is thus able to gather and add to his database the component data and routes that are relevant to the Indian marketplace. Should VMS users in Singapore require a fuel map for an engine exported out of India, they shall contact the local database administrator in India.

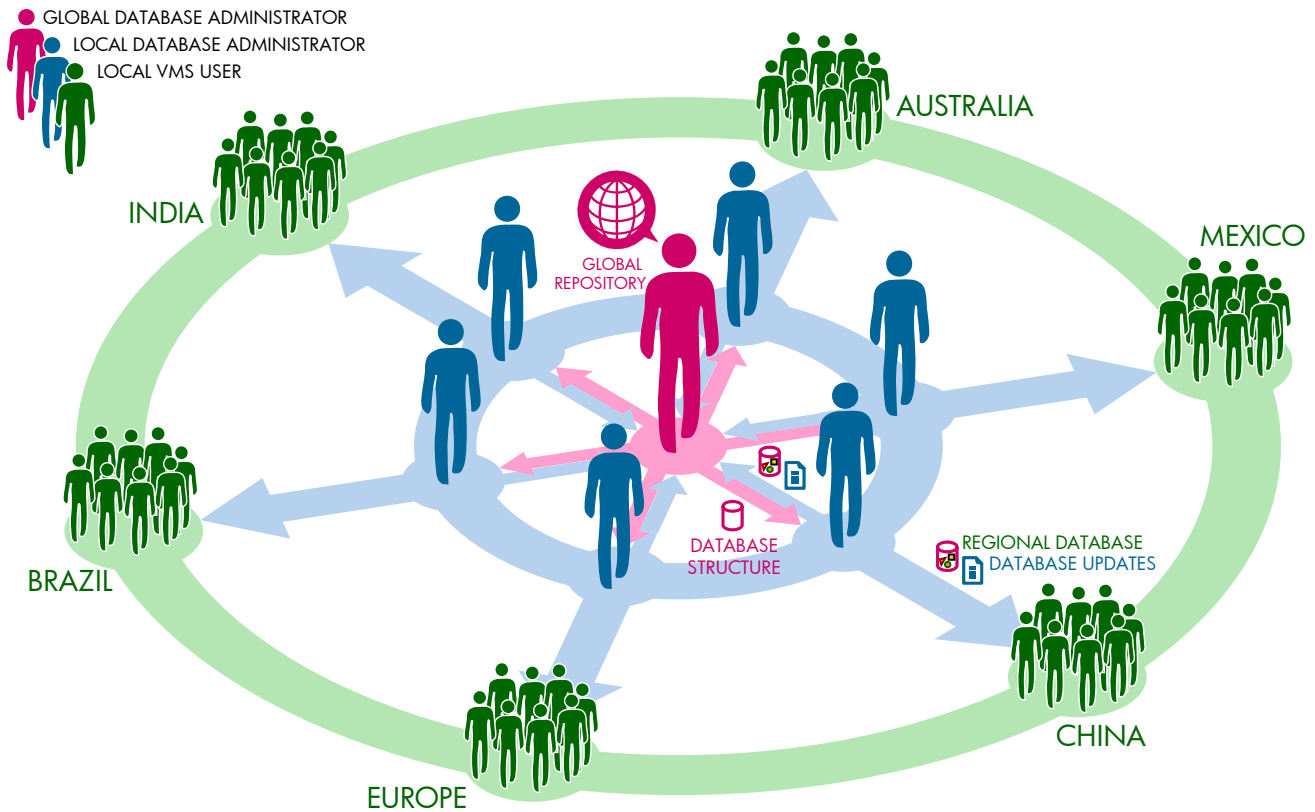


Figure 6: Decentralized VMS database governance infrastructure

At regular intervals, all local administrators share their databases both with the local VMS users and with the global administrator. The global administrator maintains a consolidated global database at all times. To help safeguard corporate information from exposure and theft, we have implemented a proprietary data masking technology to distribute all database data securely using Structured Query Language (SQL) update files; we cannot discuss this data encryption technology in this paper due to security concerns. VMS users use the GUI to securely load the SQL files and update the resident databases on their computers.

The VMS database has grown to become the de facto global repository of information for product-life cycle management of fuel maps and engine calibrations. This solved the historical problem of data dispersion and difficulties around safekeeping and accessibility of such confidential corporate information. The MySQL database has also streamlined data interchange between product development and marketing teams across the Cummins enterprise.

XML FILES FOR MISSION SETTINGS

The VMS application interface can load the definition of one or more missions from each XML file. By interacting with the project tree in the VMS interface, a user can then edit any of the vehicle or mission settings through dialog boxes, copy or delete missions and save mission definitions back into XML files. The default installation of VMS supplies a few XML templates that contain standard missions, such as line-haul trucks, transit bus, school bus, and pick-up and delivery trucks. The user can customize these templates or use these to setup other vehicle missions. XML files are now the common denominator for interchange of vehicle mission configurations among VMS users. Users can also interact with XML files in Microsoft Excel and setup multiple missions for comparative analysis and design of experiments studies.

FRONT-END INTERFACE, REPORTS AND GRAPHS

The full-featured VMS interface, illustrated in Figure 7, uses a MATLAB platform for the front-end and application logic. Once a mission simulation is complete, VMS displays a set of default reports and graphs for standard results: simulation summaries, performance summaries, route altitude profile, and engine operating duty cycles. A VMS user could also choose from a number of supplementary graphs and reports that report all aspects of engine and vehicle performance during the mission: vehicle power requirements, engine energy audits, gradeability and startability, and vehicle performance by transmission gears. A user may save reports as HTML files and export instant-by-instant simulation results as text files.

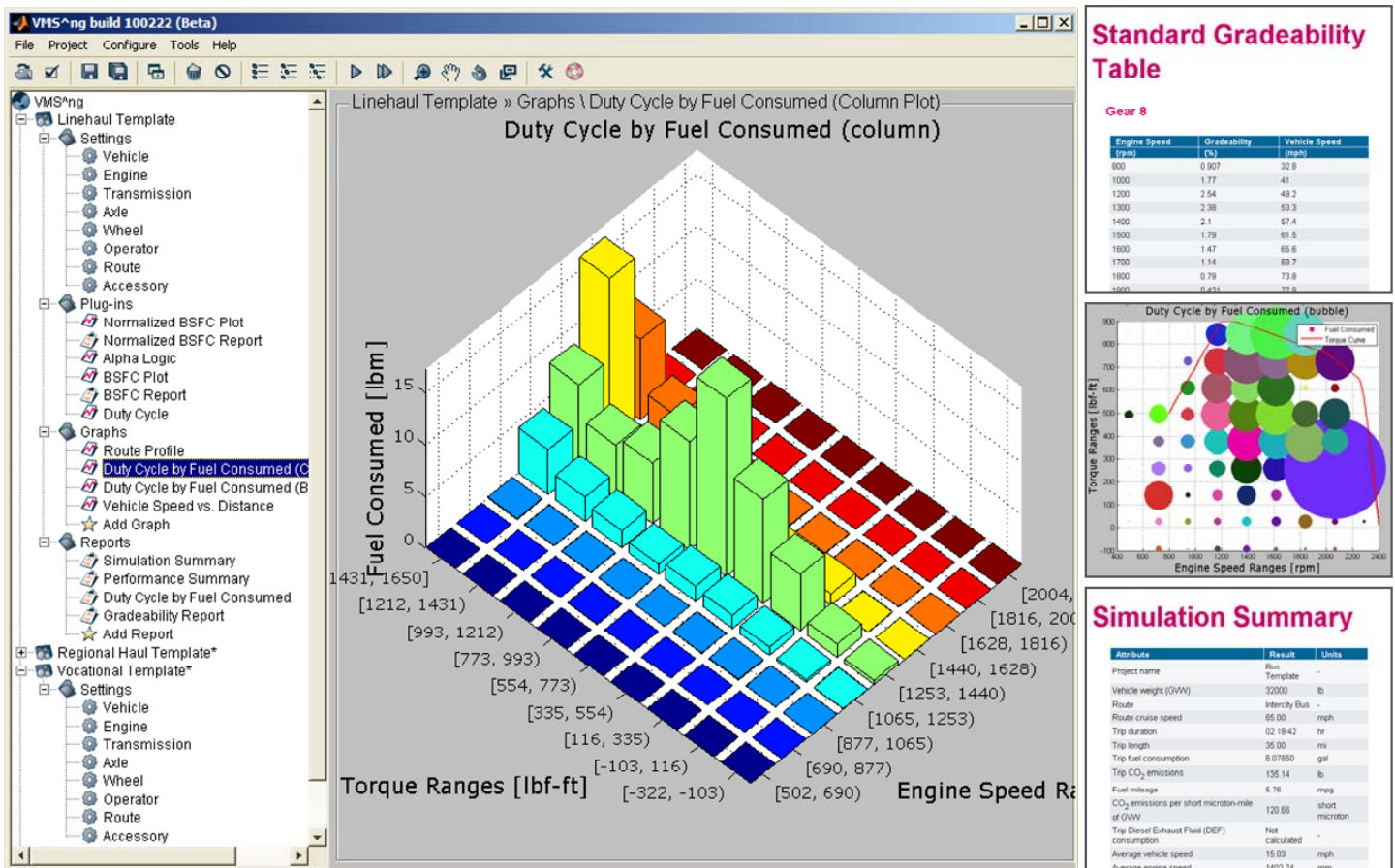


Figure 7: VMS graphical user interface with content-rich elements

BACK-END COMPUTATIONS: VEHICLE MODELING PHILOSOPHY

VMS adapts the PureSim version of the Cummins CyberApps simulation software suite that we introduced in an earlier section of this paper. PureSim imitates the real-world operation of a vehicle using the “feed-forward philosophy” [2] of vehicle modeling, as illustrated in Figure 8. Central to this system is the driver-operator who adjusts the clutch state and transmission gear. The driver also regulates the throttle based on the instantaneous requirements of road speed limit and topography. The map-based engine model then transforms the throttle command from the driver into torque. This torque output from the engine successively translates through each of driveline components and results in traction force on the wheel. The net force acting against the inertia of the vehicle calculates the instantaneous vehicle acceleration and velocity. The vehicle speed then feeds forward to the engine through the driveline components. PureSim also accounts for grade resistance, aerodynamic drag, rolling resistance of the tires, transmission and axle inefficiencies, and the rotational inertias of each of the driveline components.

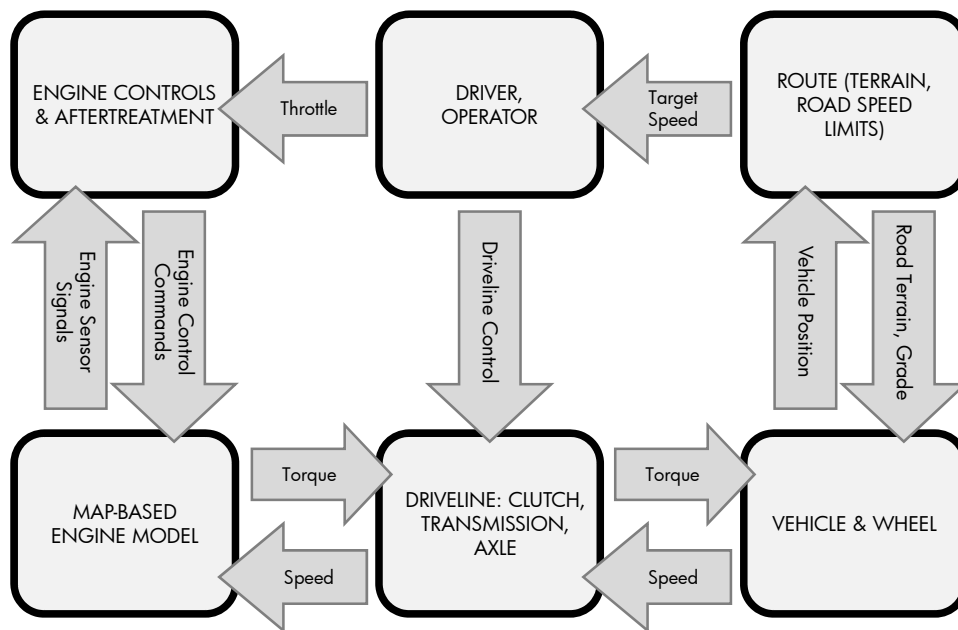


Figure 8: Vehicle modeling philosophy in VMS, adapted from the PureSim version of the CyberApps

PLUG-IN TECHNOLOGY

Plug-in modules are supplementary features that extend the technical and reporting capabilities of VMS. The plug-in technology we have devised for VMS enables us to restrict certain advanced features of the software to privileged users. For instance, engine fuel maps are highly confidential; therefore, the ability to view fuel maps is available only to users in product engineering and inaccessible to users in marketing and sales. The base VMS software that we distribute contains these advanced plug-in modules; however, they are inactive by default. Depending on the credentials of a user, we can remotely turn on the relevant plug-ins using an activation key.

The plug-in technology has also enabled us to release experimental VMS features to a select set of users and test them thoroughly before mainstream release. Using plug-ins, VMS can also host add-on MATLAB and Simulink programs sourced from third-party developers within Cummins. These add-ons use results from

vehicle mission simulations and engine data from the MySQL database to run engineering calculations and display results using standard VMS graphs and reports. For example, the tribology group has implemented, based on its empirical methods for calculating engine oil consumption using engine operating duty cycle from VMS simulation, a plug-in for estimating oil drain intervals and designing the size of the engine oil pan.

SOFTWARE IMPLEMENTATION

The VMS application is a standalone Windows 32-bit executable built using MATLAB Compiler. This compiler encrypts all the application code using the Advanced Encryption Standard (AES) cryptosystem. The application layout and graphical user interface construction both use the MATLAB based graphical user interface development environment (GUIDE). As illustrated in Figure 9, the MATLAB Java interface integrates all the rich feature content seamlessly with interactive MATLAB plots for graphs and HTML panes for tabular reports.

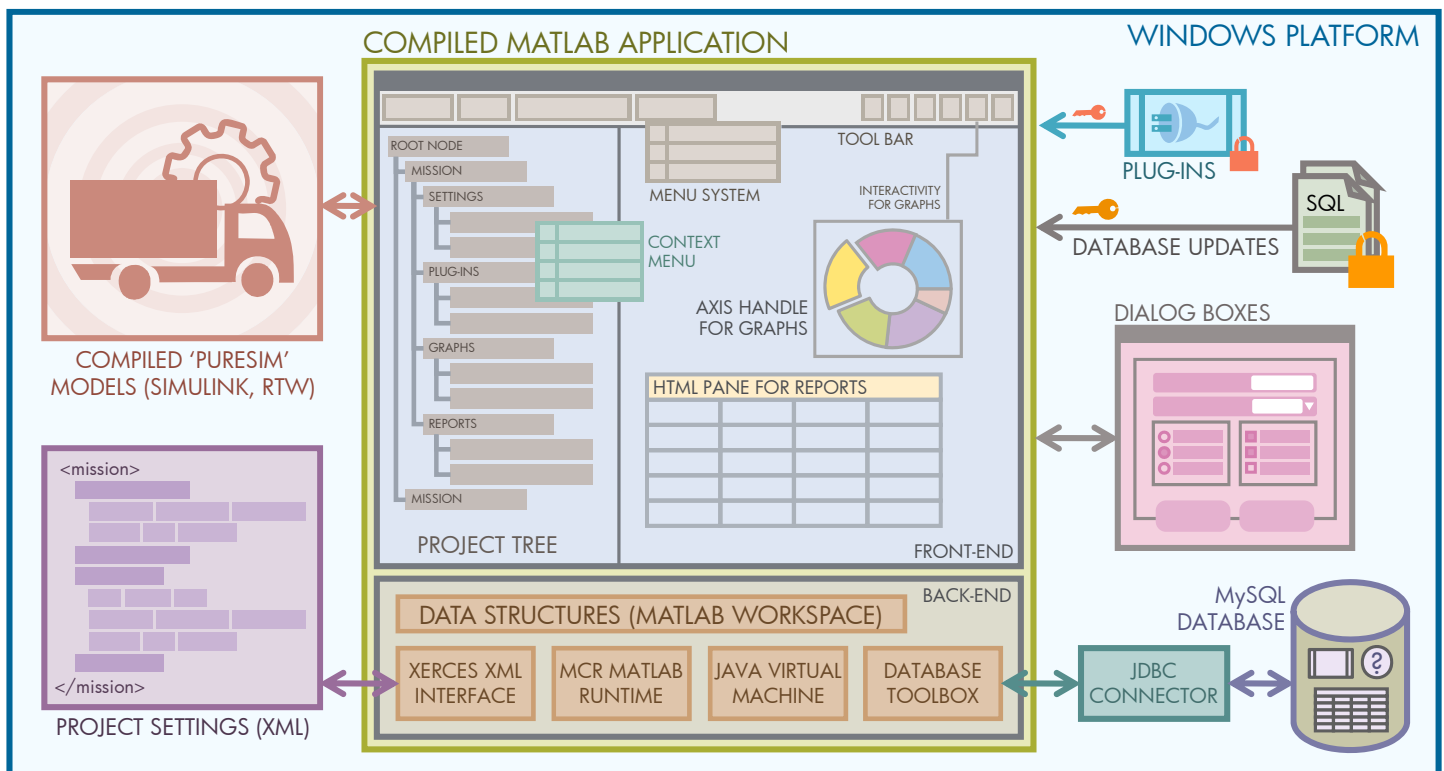


Figure 9: VMS leverages an integration of multiple technologies

VMS uses the Apache Xerces libraries for parsing, validating, serializing, and generating XML files. A database interface module uses Database Toolbox from MathWorks to connect to the MySQL database through a JDBC connector. It calls stored procedures in the MySQL database, fetches relevant data, and populates MATLAB structures that are directly accessible by PureSim simulation models. This method of using a compiled MATLAB application to access the MySQL database via an industry standard password-hashing algorithm helps safeguard the confidential data stored in the VMS database. Ordinarily, VMS just fetches data from the database. From time to time, when a VMS user decides to update his database, he gets an encrypted Structured Query Language (SQL) data file from his local database administrator. VMS can then decrypt this data file and issue SQL commands that write data into the MySQL database.

The computational backbone of VMS consists of stand-alone Rapid Simulation (RSIM) executables of Simulink models compiled using the Real-Time Workshop. Simulations complete in non-real-time; a 100-mile mission completes in 30 seconds on a standard desktop computer, for example.

INSTALLATION AND DISTRIBUTION

We deploy VMS using a setup script that installs (1) MATLAB Compiler Runtime (MCR) on computers that do not have an installed version of MATLAB, (2) the MySQL database system, (3) the JDBC drivers, (4) the main VMS application software, and (5) a set of binary Windows specific wrapper files generated by the MATLAB Compiler. The install script also starts the MySQL server on the target computer, initializes the MySQL database, updates the Windows registry, and installs icons and shortcuts in the Windows environment.

The VMS installation does not, however, populate the database. When users launch VMS for the first time, the interface directs users to contact their local database administrator to get the database relevant to their geography or function. VMS also directs users to an intranet Web site where they can download training videos.

To supplement VMS, we have also developed a collection of stand-alone software utilities to gather, process, and prepare SQL files for adding data to the VMS database. Foremost among these are (1) a software utility to process fuel maps from test cell data, and (2) a software utility to prepare topographic routes from mapping services or GPS data collected from tracing routes. All of these tools use the same architecture as the main VMS tool, drive improved data quality and promote standardized business processes across the enterprise.

APPLICATION CASE STUDIES

Since June 2009, Cummins salespeople and engineers around the globe have downloaded, activated, and used VMS to support customer needs. By far, the most widespread use of VMS is to simulate the missions of customers on predefined corporate-standard or customer-specific routes, study vehicle performance, and tailor their trucks and fleets for their most favorable outcomes.

VMS served as a critical enabler of concurrent engineering during the Cummins product development cycle (formally known as the Value Package Introduction (VPI) process) for 2010 EPA-complaint diesel engines. We developed a new corporate-standard process to guide VPI projects: all along the VPI process, performance development engineers input the fuel maps of prototype engines and calibrations from test cells into VMS to get rapid feedback on the mission-specific fuel consumption of engines. Advanced combustion research engineers studied analytically designed fuel maps using VMS to evolve strategies for better fuel economy and carbon efficiency of future products.

Below, we present four representative case studies that demonstrate the diversity of applications and the comprehensiveness of VMS as a value-based engineering tool that appeals to a wide audience within Cummins.

CASE STUDY 1: VEHICLE PERFORMANCE ANALYSIS OF A 16-YEAR OLD WRECKER TRUCK

The VMS database has developed into the de facto repository of legacy and contemporary fuel maps and vehicle parameters within Cummins, as we mentioned in an earlier section of this paper. A sales territory manager in charge of a large public transit authority recently analyzed the gradability of its 16-year-old wrecker truck. She found the 280hp, Cummins L10 engine in the VMS database along with the relevant transmission and other vehicle parameters required for her vehicle performance analysis.

When the transit authority had originally purchased the wrecker in 1994, it had been specified to tow 70,000 pound transit buses. The transit authority desired to compare how the wrecker would tow transit buses that now weigh 80,000 pounds. By using VMS, the territory manager predicted a 14 percent reduction in gradability at all vehicle speeds. For instance, at 28 mph and in gear number six, if the truck could tow a 70,000 pound-bus up a 3.62 percent grade, it could tow an 80,000-pound bus up only a 3.11 percent grade.

CASE STUDY 2: FUEL ECONOMY SENSITIVITY ANALYSIS

Fuel is one of the chief operating costs in the trucking and logistics industry. Engine and vehicle manufacturers go to extraordinary lengths to improve the fuel economy of engines and fuel mileage of vehicles. This case study presents results from a sensitivity analysis on five levers to improve fuel economy on a mass-produced full-size pickup truck powered by the Cummins 5.9-liter ISB engine. As part of cost-benefit analyses, our OEM-customer engaged us to help with estimating the impact of (1) reducing vehicle curb weight through lighter materials, (2) reducing wind resistance by means of aerodynamic aids, (3) supplying more expensive tires with lower rolling resistances, (4) installing more efficient engine accessories (the cooling fan, alternator, Freon compressor, and power steering pump), and (5) reducing the axle ratio. Results charted in Figure 10 indicate that of the five parameters we studied, the choice of a lower axle ratio and reduction of curb weight of the truck would improve the vehicle fuel economy the most. Fuel economy was least sensitive to the choice of engine accessories.

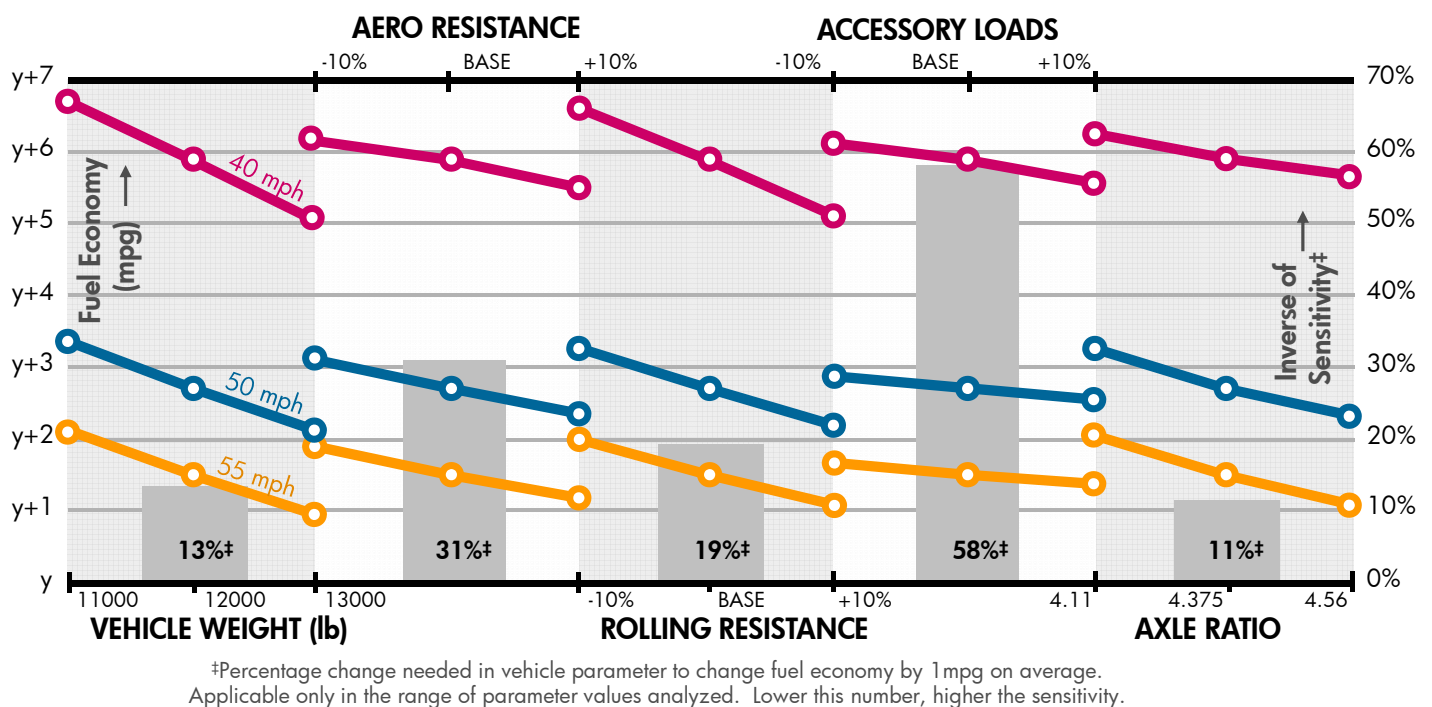


Figure 10: VMS sensitivity analysis on fuel economy of a medium-duty utility truck

CASE STUDY 3: ENERGY EFFICIENCIES FROM EXPANDING THE OPERATION OF HIGHER PRODUCTIVITY VEHICLES ON THE MAINE INTERSTATE SYSTEM

Higher Productivity Vehicles (HPVs) are six-axle tractor semi-trailers with gross vehicle weight (GVW) of greater than 80,000 pounds. Presently, an exemption to the federal limit on vehicle weights on the interstate

system allows HPVs to operate on certain sections of the interstate routes in Maine. On the non-exempt sections, HPVs are required to divert off the interstates and proceed on state roads. As part of a proposal for policy change, the Maine Department of Transportation (DOT) requested American Transport Research Institute (ATRI) to quantify the fuel consumption benefits and greenhouse gas emissions reductions from allowing HPVs to operate on the non-exempt sections of the interstate system. ATRI engaged Cummins to use the VMS software to simulate representative HPV missions and quantify these energy efficiencies.

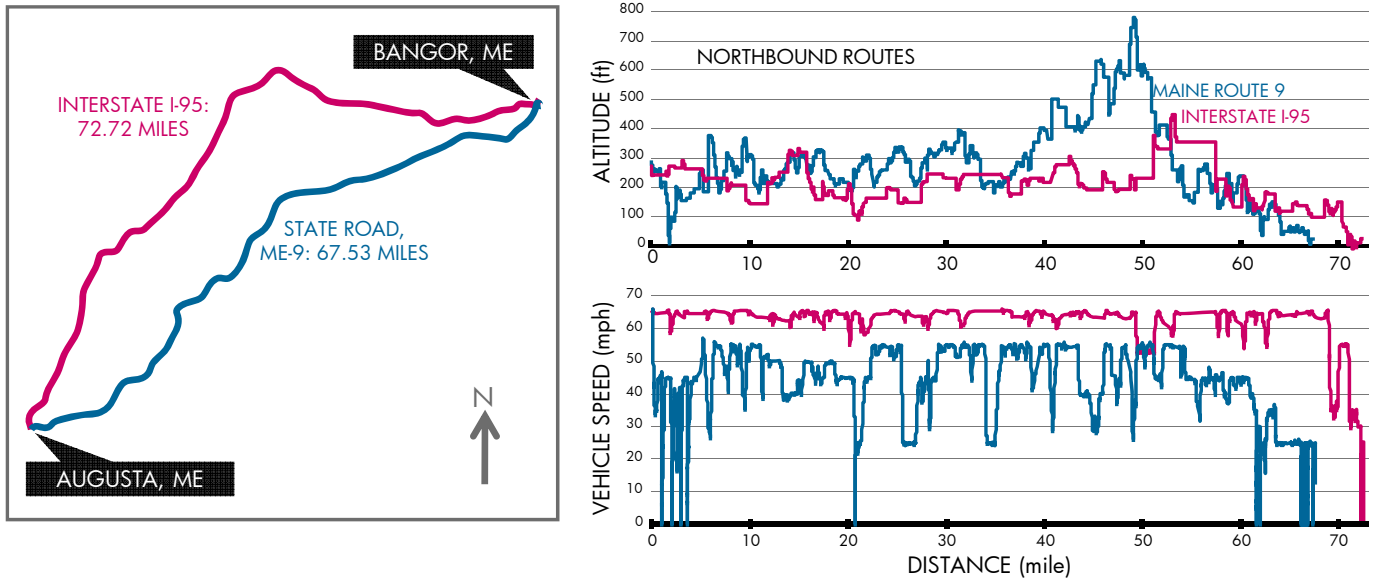


Figure 11: Specifications of the State Road ME-9 and Interstate I-95 routes

Using a mapping service, we defined a stretch of the currently non-exempt interstate I-95 connecting the cities of Augusta and Bangor in Maine and a matching route along state road ME-9. The I-95 and ME-9 routes have identical endpoints: one endpoint on I-95 near Augusta and the other endpoint on a city street in Bangor as illustrated in Figure 11. Maine DOT provided the posted road speed limits and locations of traffic signals along the routes. The I-95 route consisted of two traffic signals (both in the city of Bangor) and five changes in posted road speed limits. More than 95 percent of this route had a steady road speed limit of 65 mph. In contrast, the ME-9 route passed through several towns and therefore included 14 traffic signals and 28 changes in posted speed limits. Furthermore, the ME-9 route was 5.19 miles, or 7 percent, shorter than the I-95 route.

Using VMS software, we analyzed two extremes of traffic scenarios. The first scenario represented a best-case “no stops” condition where the HPV proceeded through every traffic signal along the routes uninterrupted. This condition modeled no deceleration or stop at each traffic signal; therefore, the HPV maintained constant speed through the traffic signal per the posted road speed limits. The second scenario represented a worst-case “all stops” condition where the HPV stopped at each traffic signal along the routes for 20 seconds. Therefore, this condition caused the HPV to decelerate from its drive speed, stop for 20 seconds at each traffic signal, and accelerate back to the posted road speed limits.

VMS simulated the performance of a typical 100,000-pound HPV with a three-axle tractor and a three-axle semi-trailer on both the northbound and southbound routes with the above two traffic scenarios. A 485 hp, 1,650 lb-ft Cummins ISX engine with EPA 2007 certification powered the HPV. VMS predicted that, in spite of the fact that the I-95 route was 7 percent longer than the ME-9 route, the HPV could expect the following benefits from using the I-95 route instead of the ME-9 route: (1) 28 to 32 percent savings in trip times, (2) 7 to

11 percent savings in diesel fuel consumption, and proportional reduction in carbon-dioxide emissions, and (3) 4 to 8 percent reduction in particulate emissions and mono-nitrogen oxide (NOx) emissions. Nevertheless, the engine produced on an average 33 percent more power on the interstate I-95 route because of the higher average road speeds on the interstate.

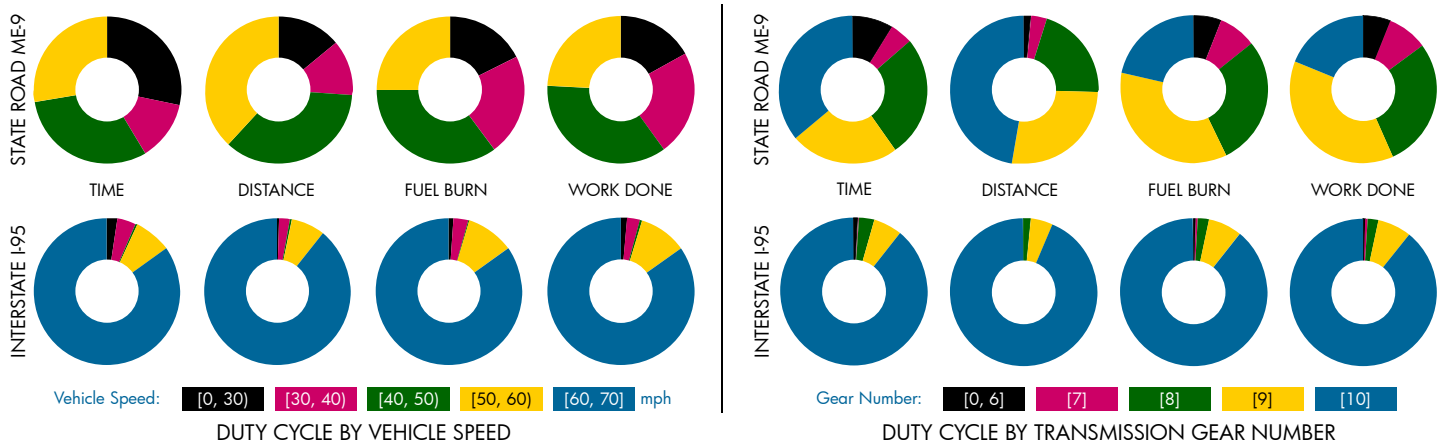


Figure 12: Comparisons of duty cycle splits on the northbound ME-9 and I-95 routes

For a more detailed comparison between the I-95 and ME-9 routes, consider, for example, the case of the HPV operating in the northbound direction and stopping at every traffic light for 20 seconds. Figure 12 illustrates the differences in resultant duty cycles by vehicle speed and transmission gear number. Figure 13 characterizes the engine operating duty cycles in terms of time and fuel.

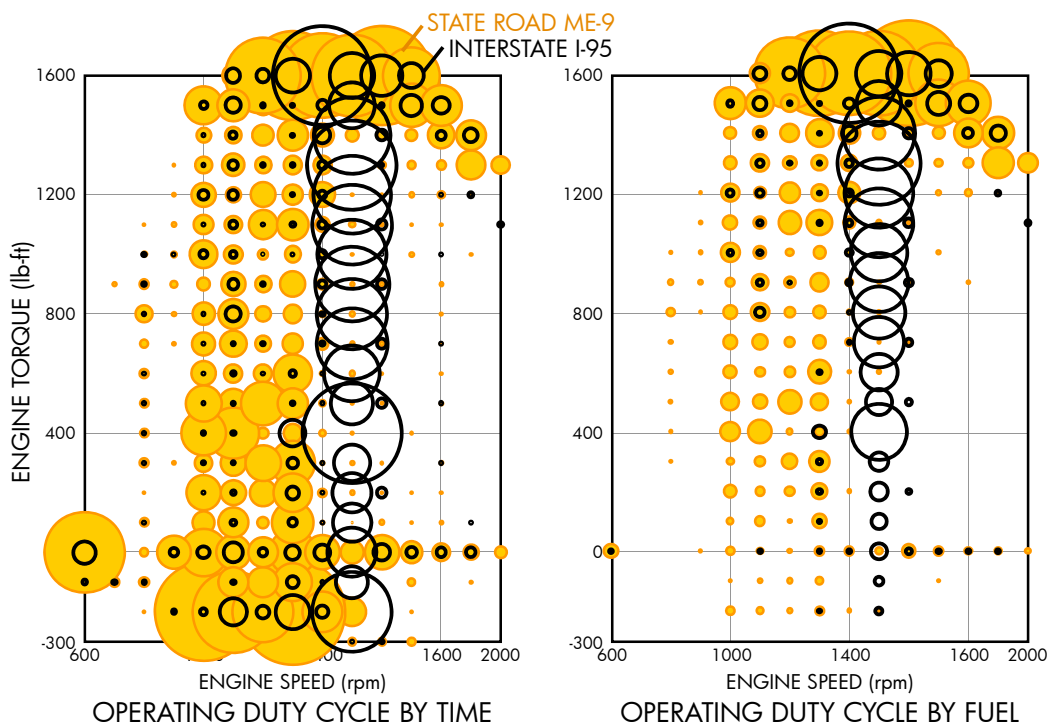


Figure 13: Comparisons of engine operating duty cycles on the northbound ME-9 and I-95 routes

Figure 14 illustrates the comparative fuel energy audits between the interstate I-95 and state road ME-9 routes. We observed that: (1) The sums of energy consumption by driveline, rolling resistance, aerodynamics and braking were within 6 percent between the I-95 and ME-9 routes, (2) Aerodynamic losses were 86 percent greater on the I-95 route owing to higher road speeds on the interstate, (3) Braking losses were 460 percent greater on the ME-9 route due to traffic and varying road speeds, and (4) Tire rolling resistance losses were 12 percent greater on I-95, the 5.19 mile (7 percent) longer route. On the I-95 route, the engine produced 33 percent more power and did 8 percent less work. The 11 percent savings in fuel consumption comes primarily from doing lesser work and operating the engine in areas of lower fuel consumption on the engine fuel map. The power requirements graph, also in Figure 14, illustrates that, above vehicle speeds of 60 mph, aerodynamic drag is the most significant contributor to the vehicle power requirements.

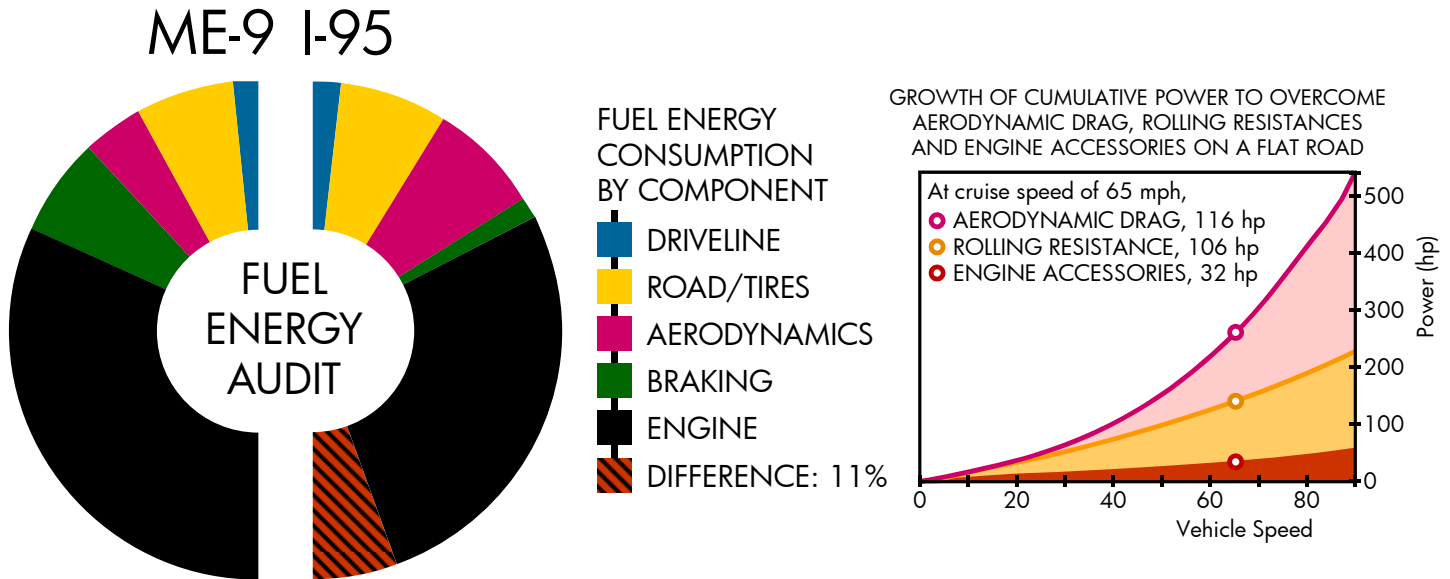


Figure 14: Comparative fuel energy audit records between the northbound ME-9 and I-95 routes

Based on the routes we modeled, we concluded that expanding the exemption to the federal limit on vehicle weights on the entire Maine interstate system could result in significant economic benefits from lower diesel fuel consumption, reduced greenhouse gas emissions, and lower particulate emissions, not to mention of other potential benefits such as the reduction of congestion in the towns along the state road, which were outside the scope of our analysis. ATRI presented these results to Maine DOT in September 2009 [4].

CASE STUDY 4: FUEL ECONOMY IMPROVEMENTS FROM FINE-TUNING VEHICLE CONFIGURATION FOR A LARGE TRUCK FLEET

Application engineers across the Cummins distributor network often use VMS to help truck fleets “spec” their trucks for their particular missions and choose a gearing (choice of transmission, axle and wheels) that makes the engine run at the ideal engine speed for best fuel economy.

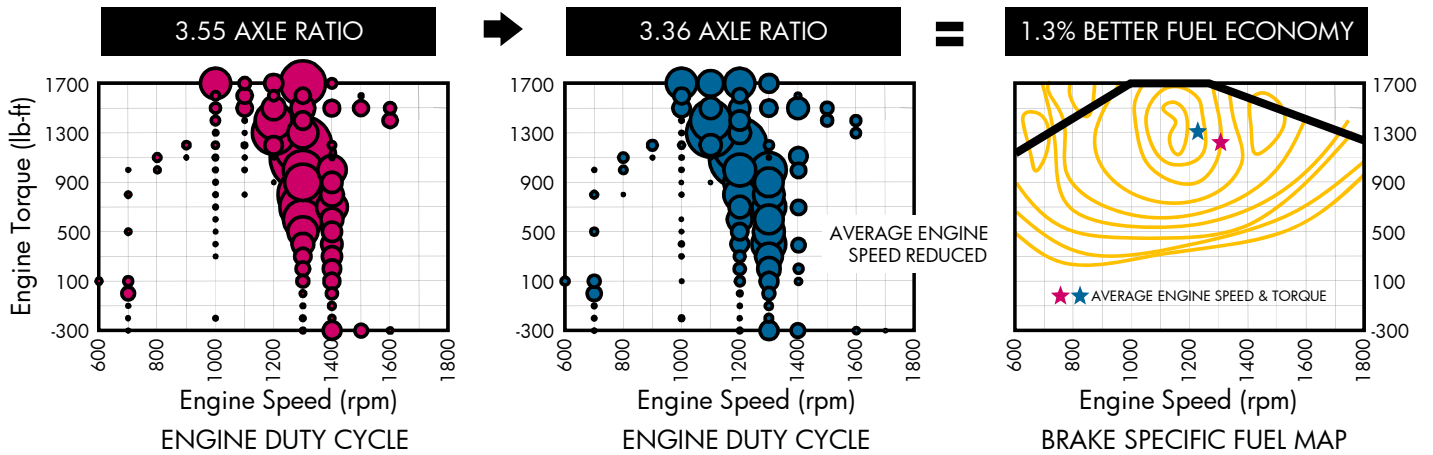


Figure 15: Reduction in axle ratio results in lower fuel consumption

The ability to construct virtual routes that represent typical customer missions using mapping services or GPS traces helps VMS calculate whether customer trucks could ascend mountainous roads with steep grades. After running a mission simulation, VMS can chart the engine operating duty cycle against its fuel map. VMS users can then use this comparison to determine better vehicle gearing for improved fuel economy, less frequent transmission gear shifting, and better vehicle performance. Figure 15 illustrates the fuel economy advantage of dropping an axle ratio on a fleet of line haul trucks. Using VMS, an application engineer at a Cummins distributor demonstrated that the average engine speed would decrease by 75 rpm, causing the engine to operate in regions of lower fuel consumption on the engine fuel map.

FUTURE WORK

In the near term, in addition to supporting application projects such as the ones we described above, the VMS team is improving the technical capabilities of VMS. We are working with worldwide customer engineering teams to add technical features and plug-ins that can broaden the reach of VMS. We are also developing a one-day course on the basics of vehicle engineering and VMS analysis to instill systems thinking and help Cummins performance engineers appreciate the broader context of their work.

In the long-term, we see a robust pipeline of large initiatives to help improve the fuel and carbon efficiencies of Cummins engines and deliver better products for our customers:

- The increasing demand for fuel resources, rising fuel costs, and concerns over global climate and ecology have drawn considerable attention to lowering vehicle fuel consumption and reducing greenhouse gas emissions. The ability to analyze rapidly the engine and the vehicle as a system makes VMS a key enabler of Cummins research on further improvements to develop and deliver means of better energy efficiency [3]. A significant set of energy efficiency analyses needs is headed our way.
- The ability of VMS to optimize the vehicle configurations around the missions of our customers opens up a multitude of opportunities to develop computer heuristics to suggest means of improving vehicle fuel economy. VMS will be the keystone for such next-generation optimization tools. Using VMS as the core-computing element, we are currently building a high-performance grid computing system for significant compute power. This service shall be available to Cummins research and product engineering teams as software as a service (SaaS) to assess and communicate better product-to-customer fit.

CONCLUSIONS

In this paper, we described the significant business purposes, software architecture, and applications of Vehicle Mission Simulation (VMS,) the Cummins proprietary value-based engineering tool for engine-vehicle integration. Using proven technologies, we have equipped our sales force and engineers with the ability to leverage technical information to provide better business value to Cummins customers.

VMS is now part of the standard selling process across the Cummins enterprise; our users have already realized significant advantages from the ability of VMS to model the specific missions of our customers, study engine and vehicle performance and improve the fuel and carbon efficiencies of Cummins engines. They have used these results to deliver solutions that drive better business outcomes for our customers and have thus proven many of the implied benefits of VMS. The growing utility of VMS and its comprehensiveness is a very strong validation of the product placement and technology strategies that we have executed.

REFERENCES

1. Guzzella, L., & Antonio, S. (2005). *Vehicle Propulsion Systems*. Berlin Heidelberg: Springer-Verlag.
2. Seger, J. (2009). *Framework for the Regulation of Greenhouse Gases from Commercial Vehicles*. 2009: Cummins Inc.
3. Tunnell, Michael. (2009). “*Estimating Truck-Related Fuel Consumption and Emissions in Maine: A Comparative Analysis for a 6-axle, 100,000 Pound Vehicle Configuration*”, Arlington, VA: American Transport Research Institute.

CONTACT INFORMATION

Nagesh Belludi is the product manager for VMS. Jeremy Raymond and Joshua Receveur are the VMS software architects. The VMS team is part of the Customer Engineering group at Cummins and may be reached at Cummins Inc., 1900 McKinley Avenue, Columbus, IN 47201 or via e-mail at nagesh.y.belludi@cummins.com

Arvind Hosagrahara is a senior technical consultant at MathWorks. He may be reached at MathWorks, 39555 Orchard Hill Place Suite 280, Novi, MI 48375 or via e-mail at arvind.hosagrahara@mathworks.com

ACKNOWLEDGMENTS

The VMS team is grateful to Gary Salemme and Burit Kittirungsi of the System Performance Analysis team at Cummins for their support in integrating the CyberApps simulation tools into VMS and to Bill Kendrick of the Vehicle Integration team at Cummins for his expertise in vehicle performance and gearing. Our thanks are also due to Michael Tunnell, Director of Environmental Research at the American Transport Research Institute and to the Maine Department of Transportation for the opportunity to analyze the energy efficiencies from allowing higher productivity vehicles to operate on all of the Maine interstate system.

VMS is a trademark of Cummins Inc. MATLAB, Simulink, and Real-Time Workshop are trademarks of The Mathworks, Inc. JDBC, JVM, Java and MySQL are trademarks of Oracle Corporation. Windows is a trademark of Microsoft Corporation. All other product or brand names mentioned in this paper are trademarks or registered trademarks of their respective holders.