FALL 2019

KEYnote 38

THE WIBU-MAGAZINE

IP Protection in Additive Manufacturing

Highlights

- CodeMeter & Docker
- IP Protection for Software without Licensing
- CodeMeter Cloud Pilot Phase: The Birth of a New Product

WIBU SYSTEMS

Content

Dear Clients and Partners!

Germany's business weekly, Wirtschaftswoche (www.wiwo.de) has recently reminded its readers that "The label ‚Made in Germany' continues to wow customers worldwide" based on a survey by polling institute YouGov and Cambridge University. This makes us proud but reminds us that it is also a constant challenge to always reach for the top and exceed expectations. History waits for no-one, and China shows us how large public building projects get finished on schedule, trains arrive on time, and AI and Big Data applications produce results.

We are working intensively with partners from all corners of the world: Africa, the United States, Japan, China, and elsewhere. An exciting project researching 3D printing has just been launched with Beijing University, the RWTH Aachen, and German and Chinese businesses. We have been granted two patents, for our pseudolink and the Blurry Box protection technology. Both are already integrated into our CodeMeter Protection Suite, where they are helping us protect your applications from reverse engineering and product piracy.

We know the value of free trade, and we believe in the innovative minds of all our people around the globe. That is why we are investing in a new head office and our very special House of IT Security, the future home of researchers and entrepreneurs in our field.

Enjoy this issue of the KEYnote magazine and read about exciting news and use cases: See CodeMeter in the Azure cloud; follow us as we take our first steps with CodeMeter Cloud, our upcoming solution for high-performance and high-scalability CmContainers in the cloud; learn more about CodeMeter in Docker environments, the new IP Protection mode, 3D printing applications, and how CodeMeter compares to the much-vaunted Blockchain for software protection and licensing.

I am looking forward to seeing you again – be it here in our headquarters in Karlsruhe, at a Wibu-Systems seminar, or at one of the upcoming fall shows, like COMPAMED and SPS in Germany or the ET in Japan.

Best regards from

Oliver Winzenried

CEO

CodeMeter SmartBind & Microsoft Azure

Virtual environments make it difficult for activation-based licenses to be securely bound to the machines they are meant for. The properties that would normally be used as a fingerprint of the target system are often generic in virtual machines, and a change to these properties (often happening in bulk in data centers) has the potential to break an entire collection of licenses in one fell swoop. Wibu-Systems has worked with Microsoft to develop a much better approach for virtual systems running in Microsoft Azure.

CmActLicenses are activation-based licenses that need no separate hardware, as they employ a signed and encrypted license file. The unique encryption technology allows the license file to contain symmetric and asymmetric keys on the user's computer itself, which can then be used for the various cryptographic operations happening in CodeMeter.

CodeMeter SmartBind is Wibu-Systems' patented solution for binding CmActLicenses to their target devices. CodeMeter SmartBind creates a fingerprint of the user's computer by referencing different traits and properties, each with their own specific weighting, like the hard drive, motherboard, or CPU. A special and similarly patented tolerance mechanism makes sure that the CmActLicenses and the keys stored in them remain valid even if the user replaces parts of their computer's hardware. The fingerprint evolves automatically to match the environment and operating system preferred by the user.

Creating such a unique and uncopiable fingerprint for binding licenses to virtual machines still remains a tough proposition. After all, freedom and flexibility to experiment with simulated hardware properties is part of the raison d'être of virtualization. Wibu-Systems has continued to refine the inner workings of CodeMeter SmartBind to enable strong binding for CmActLicense even in virtual environments.

Teaming up with Microsoft in a project for a shared client offered important new insights. Microsoft Azure data centers have the opportunity to use a web service to access certain parameters of the Azure environment. For the project, a volume license for the client was given a custom binding by reading out the licensing details of Microsoft Azure itself. This makes the same binding work for all installations of the client's software operated under the same subscription in the Microsoft Azure environment.

To bind licenses directly and individually to virtual machines, Windows systems operating in Microsoft Azure now use a separate ID provided there. This ensures that the binding remains intact even if the virtual machine's properties are changed. The system would recognize a cloned machine (or, more precisely, the integration of a new disk image in a virtual machine), and the license would be broken.

The new recipe for fingerprints using the special Microsoft Azure ID is employed for all CmActLicenses on Windows systems first created with CodeMeter Version 6.90 or later. All CmActLicenses created by earlier versions of CodeMeter 6.90 continue to use the established format. In order to benefit from the new system's advantages, the CmActLicenses would have to be replaced; to do so, the activated licenses could be returned to CodeMeter License Central, the empty CmContainers deleted, and new licenses activated.

The next version of CodeMeter is expected to bring the new type of fingerprint to Linux systems in Microsoft Azure upon its release in December 2019.

This solution makes CodeMeter SmartBind more robust than ever before and safe from misuse and manipulation in Microsoft Azure.

CodeMeter & Docker

Container systems like Docker are becoming an increasingly popular choice for running applications in isolation. Containers are simple to use and easy to duplicate. Software developers have a legitimate interest in protecting and securely licensing their applications even in such container environments. This article reveals how the ability to use, copy, and multiply containers can be reconciled with the wish to license software correctly, and how Docker can cooperate with CodeMeter.

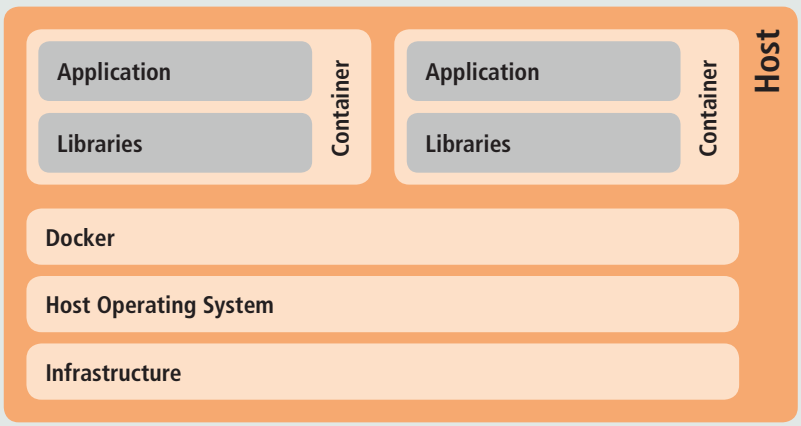
The idea of operating software in containers is not new. The basic functionality has long been part of the Linux operating system, and the Docker project has put a user-friendly container solution into the hands of the masses. Docker has reached critical mass and established itself as the leading provider of container technology. Since Microsoft decided to integrate Docker's containers into Windows Server 2016 and Windows 10, the world has come to recognize Docker as the de-facto standard in its field.

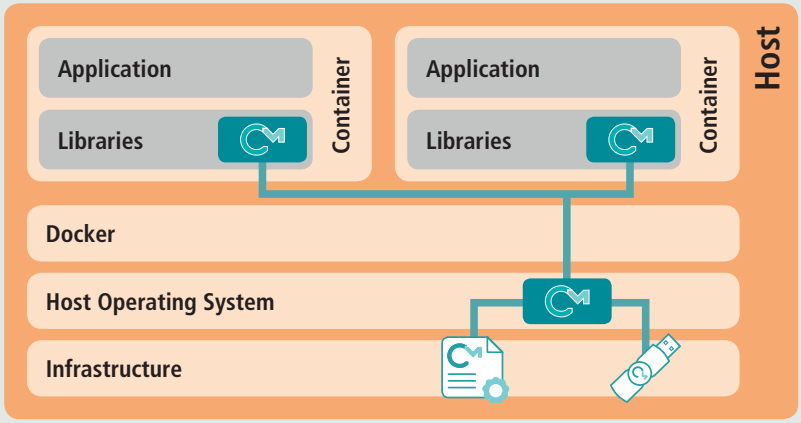
There are many benefits to running applications in a container. In essence, containers are small virtual machines (VMs) without their own operating system. They share the host system's OS kernel and important system files, making them much leaner than full-blown VMs providing the same services. This makes them cheaper to run, as the setup uses fewer resources and fewer systems need to be maintained (with updates, security patches, etc.).

Containers do not have to load the operating system, resources, or libraries on startup. Instead they can directly access the components and data of the operating environment. With no operating systems having to be installed, additional containers can also be added much faster. When the process is automated, the setup scales perfectly – one reason why containers are becoming more and more popular.

Despite all of this good news, one must not ignore the drawbacks. Compared to traditional virtualization technologies, the individual containers are not as well-isolated from each other or their host. The processes running in the containers also share the same system libraries and the same kernel, which can cause compatibility issues. Vulnerabilities or software bugs can pose real problems when using containers and might affect the system in its entirety.

How does CodeMeter behave in this environment, and what do software developers and license vendors have to remember? How can





you use containers, but keep in control of your licenses? From the brief overview above, one can see how easy it would be to copy a license created and bind it to a container. One installation of a CodeMeter license server in one container would essentially not differ at all from another one in another container. The binding properties would be identical. In order to run multiple installations of the CodeMeter license server in separate containers on a single host, several specific changes were needed: In CodeMeter 6.90, two areas needed to be changed that influence operations in a container.

The first concerns the types of CmActLicenses that are allowed to operate in a Docker container. CodeMeter Version 6.90 generally prohibits their activation and use – with two specific exceptions. The first are licenses without a concrete binding to the system hardware (NoneBind) that allow multiple imports (Reimport). These licenses serve only as a means for decrypting applications, but without any license restrictions at all, in essence forming a "Protection Only" use case. In the future, this use case would be replaced by the new IP Protection mode in CodeMeter Protection Suite (see this issue of KEYnote for more details). Since this type of CmActLicense has no license restrictions, there is no risk of fraud when the same license is used in multiple containers. The second exception are CmActLicenses specifically approved for this use. Software developers can set a new option for CmActLicenses that allows them to be activated in a container. The command line option in CmBoxPgm is „-lopt:container", usable only for Universal Firm Codes.

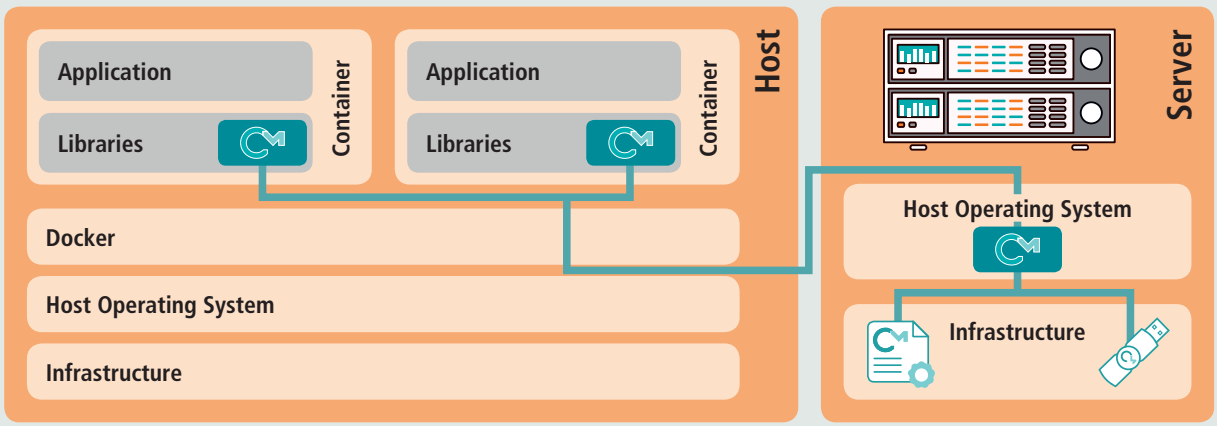
The second area concerns a change in how binding works with network interfaces. In order to reduce the possible exposure to possible attacks – not just for containers, but for all licenses – the CodeMeter license server only binds with Port 22350 if it has been configured to operate as a CodeMeter server on the network. In its basic configuration, the CodeMeter license server only binds with the localhost adapter. If it is operated in a (Docker) container, it would only receive queries from running applications and can respond either with a license present in the container (see above for the allowed types of CmActLicenses) or pass them on as a client to another CodeMeter license server on the network.

CodeMeter is versatile enough to cover a range of use cases in this manner. What they all have in common is that the container has to be operated in bridged mode – which is the standard setting out of the box.

"Protection Only": The license can be activated in the container to serve purely as a means of encrypting and using applications. The container can be scaled easily e.g. to run the application in several installations at once.

"Licensing on the Host": The license is activated on the host to restrict and keep check over the legitimate use of the application, e.g. as a CodeMeter SmartBind license. Each container has one CodeMeter license server running in client mode, which receives the requests from applications in the container and passes it on to the CodeMeter license server on the host. The server search list in the Docker network records the IP address of the host, e.g. 172.17.0.1, which will be permanent from a container perspective.

"Licensing via the Network": This use case differs from the previous one only in the location of the server. It requires a CodeMeter license server operating on the network to provide the available licenses. In each separate container, a CodeMeter license server in client mode will again receive queries from applications in the container and hand them on to the network license server.



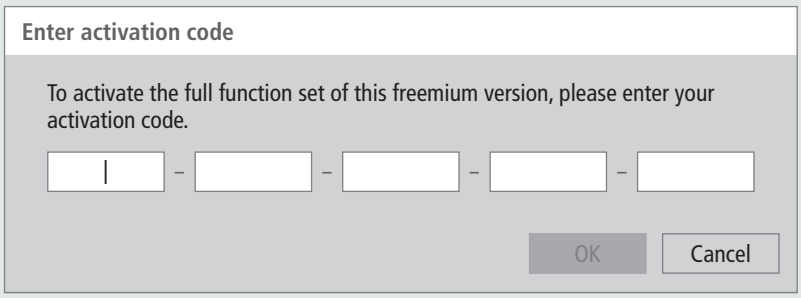
IP Protection for Software without Licensing

Wibu-Systems introduces the ideal solution to protect trial and demo software or enable freemium models with free software designed to be expanded, enriched, and leveled up with in-app purchases. IP Protection becomes easier to use with CodeMeter Protection Suite both in CodeMeter Runtime and CodeMeter Embedded.

IP Protection, the newest feature of CodeMeter Protection Suite, makes finding the right combination of software protection and licensing easier than ever before. IP Protection currently supports Windows, Linux, macOS, and Android as well as x86 and ARM processors. The new mode can be used for executable files and libraries (Dynamic Link Libraries, Shared Objects, and Dylibs).

One suite – Several tools

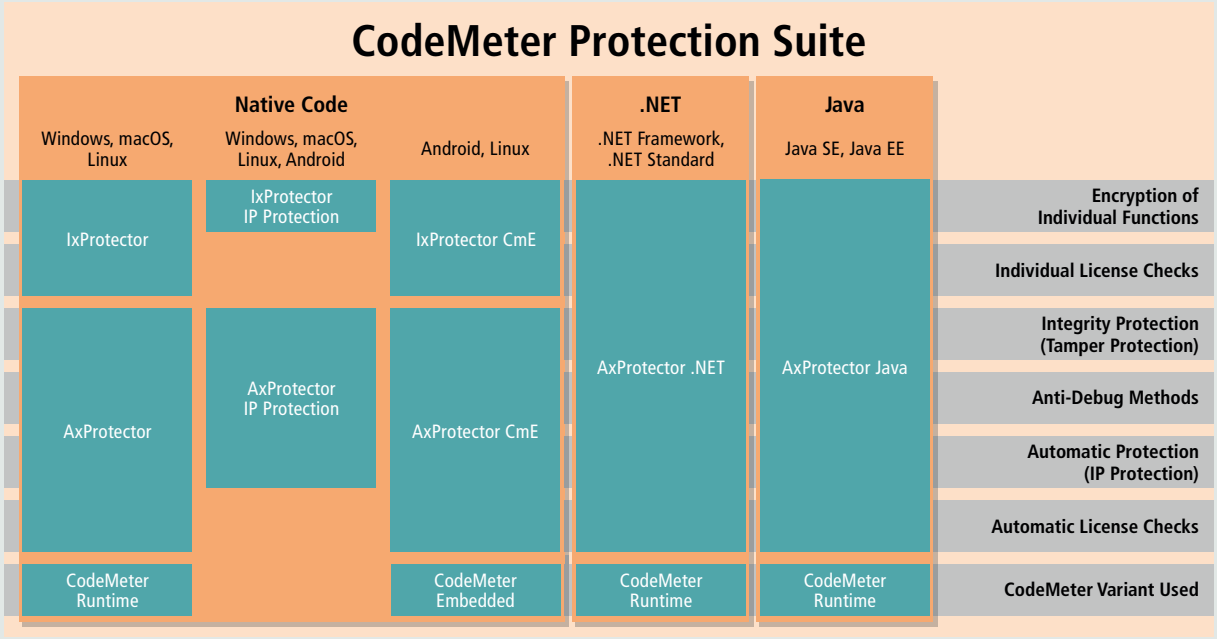
CodeMeter Protection Suite offers a complete battery of powerful tools: AxProtector is used to encrypt complete executable files, which are then decrypted with the AxEngine attached to the encrypted file. As this happens in memory, the decrypted version of the file never appears on the drive, making reverse engineering impossible. IxProtector can be used to protect individual functions in an executable, which are kept encrypted even after they are loaded into the computer's memory. They are decrypted either automatically when they are called up or in response to a special API call. This protects the software not only from reverse engineering on the hard drive; it keeps its guard up even when the attacker uses a memory dump. AxProtector and IxProtector work with both regular CodeMeter-type licenses and the new IP Protection mode. AxProtector, IxProtector, CodeMeter licenses, and the IP Protection mode can even be mixed and matched at random. The IP Protection mode for AxProtector .NET is also available with CodeMeter Protection Suite 10.50.



No runtime components required

Applications secured with the IP Protection mode need none of the runtime components of CodeMeter, i.e. neither CodeMeter Runtime nor CodeMeter Embedded, nor do they need the user to have a license ready. The decryption keys are hidden in the applications themselves to stop them from being extracted via memory dump. In this way, the IP Protection mode can be used in many possible scenarios, from firmware on embedded devices to office software, or applications running on servers or the cloud.

CodeMeter Protection Suite IP Protection prevents reverse engineering but does not stop



users from copying applications outright – a feature, not a bug, when the specific use case of free trial versions or freemium models are concerned. With the encryption working without needing a license, the application is practically self-contained, making this type of protection a very user-friendly choice indeed.

Freemium application in 3D printing

A typical example would be software for processing 3D printing data: The application is delivered with the full feature set, but artificially restricted to producing objects with a maximum size of 20mm. The users can download the software, test its features, copy it freely, and even print miniature products. However, as the entire application is fully encrypted, they would not be able to remove the restriction in order to enjoy the full potential of the software. Payment – be it by traditional means or via an in-app purchase – is the gatekeeper for getting the full, unrestricted functionality. This is where CodeMeter Runtime or CodeMeter Embedded can come into the picture. The users are given the right licenses, stored on a CmDongle, a computer-bound soft CmActLicense, or a user-bound CmCloudContainer. The license in the CmContainer deactivates the restriction in the free software, be it an artificial size limit or the ability to save, print, or export works (or whatever restriction on the full use could be imagined). This is done with CodeMeter Core API that allows the software developer to use specific functions in the CodeMeter licensing system, not just simple license checks, but also specialized options like time or volume-bound licenses or the assignment of network licenses. The developer could even integrate the transfer of licenses into the CmContainer via the API, e.g. to automatically retrieve licenses from CodeMeter License Central upon purchase from an online store or in-app purchase.

Combination with existing licensing solutions

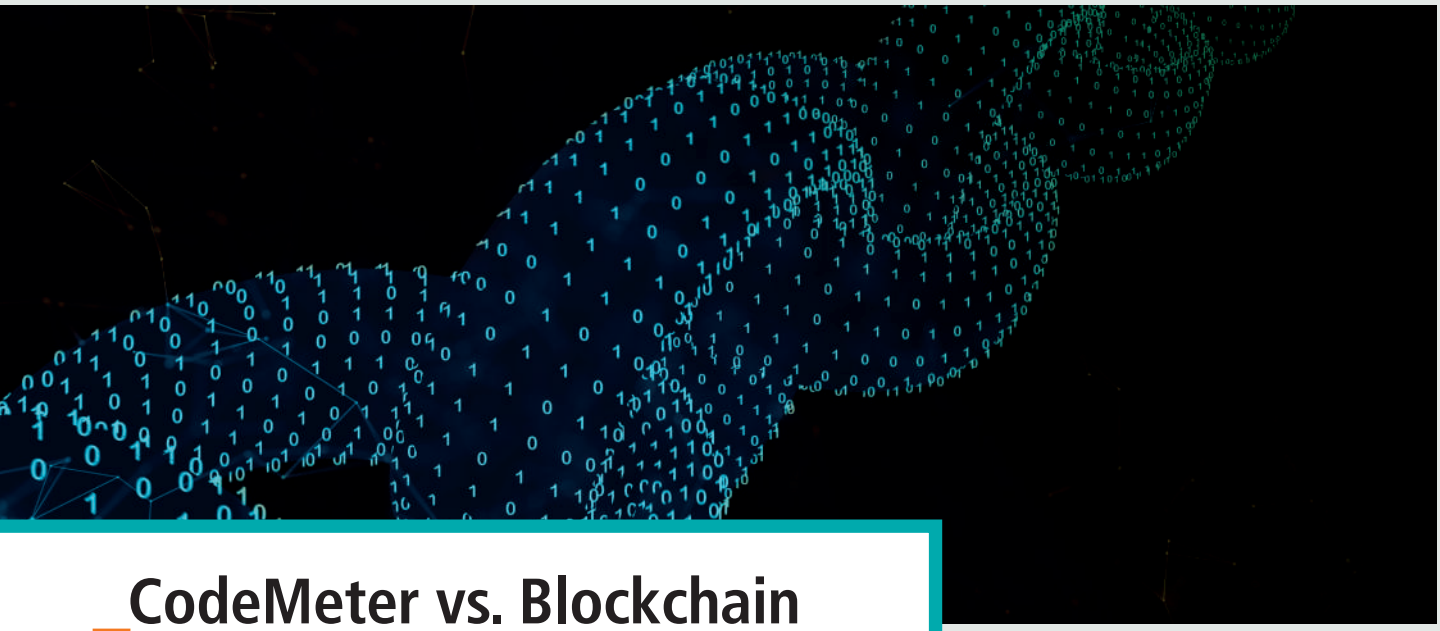
There are opportunities for other use cases, such as combining an established licensing system with the reliable IP protection capabilities of CodeMeter. Imagine a software developer who already has a homegrown licensing setup in place. However, the developer's work, the binaries, are sold to users in unencrypted form, making them easy prey for manipulation and reverse engineering. Solutions of this type that forego encryption and signature checks make it relatively easy to "patch away" the licensing checks – to manipulate the code to either skip the entire check or to keep working with a standard value. This circumvents the whole point of the licensing system. CodeMeter Protection Suite IP Protection naturally also works with other purely software-based, commercial protection systems; it can raise the bar significantly for any attempted breaches or reverse engineering. This is a favorite choice for developers migrating to the full CodeMeter setup, but still want their old licensing systems in place for a while.

Continuous development and improvement

Most users of CodeMeter will be familiar with protection only licenses. They work not unlike CodeMeter Protection Suite IP Protection, but require a version of CodeMeter Runtime installed with admin rights. This could pose a problem for B2B users working with trial software or some freemium offerings.

CodeMeter Embedded sometimes encounters version conflicts when using a combination of CodeMeter Core API and CodeMeter Protection Suite. The IP Protection mode has stepped into the breach to simplify the situation: For high-security use cases, where licenses would be kept on a CmDongle, it is still possible to use CodeMeter Core API in combination with CodeMeter Protection Suite with license-bound encryption.

CodeMeter Protection Suite IP Protection makes protecting software and introducing new business models a simple and pleasurable experience for software developers and their users alike. It was designed from the outset to work across PC and embedded platforms and to scale alongside the capabilities of CodeMeter Core API and the needs of developers. As loyal fans have come to expect from Wibu-Systems, all CodeMeter modules are fully compatible with each other and deliver an optimal, fully rounded software protection and licensing experience.



CodeMeter vs. Blockchain

Germany's federal government recently released its "Blockchain Strategy" to great fanfare. Judgement is still out on whether this strategy will become part of the great IT success story of the new federal ID or the electronic patient's card. One thing is sure: Blockchain has become such a hype that even the slow-moving world of federal politics is taking note. No wonder, then, that more and more software developers or other owners of digital assets, such as the IP in 3D printers, are asking us about Blockchain and its potential.

Does CodeMeter use Blockchain technology?

CodeMeter is a DRM system for software and digital contents tried and tested by millions of users since its launch in 2002. By comparison, while research into the cryptographic protection of blocks has been going on in some form since 1991, Blockchain has only recently become a practical, viable technology. This makes CodeMeter its older brother. It also relies on related cryptographic processes and even uses mini-block technology in some aspects: Both technologies developed in parallel and share some family traits from their origins.

What is Blockchain?

The special idea behind Blockchain is that the data (all data) is not kept at one central location like a bank vault but spread out across the Internet on a so-called distributed ledger on many computers.

A member of the chain can then enter a transaction into the ledger. Since all other members on all links of the chain have to have the same Blockchain, the end product is an unalterable consensus log.

Data in Blockchain cannot be altered at a later point, making it essentially forgery-proof.

The data is also visible to all members, making the Blockchain transparent. There is also an option to encrypt data, but this is not the standard practice.

How do we establish consensus?

One very popular consensus method is called the "proof-of-work". It relies on solving a cryptographic operation that needs a certain amount of time. For cryptocurrencies like Bitcoin, so-called "miners" do this job. After the task has been accomplished, the new block is added to the chain. If there are several Blockchains to choose from, the longer chain wins the race. A miner who holds more than 50% of the computing power in the chain could, in theory, manipulate the chain after the fact.

Solving cryptographic tasks is a computing-intensive challenge. What makes it even worse is that many miners will be working in parallel and only the first past the post will have the right to add a new block. This type of consensus stands on very shaky ground from an environmental standpoint, as it wastes masses of energy by design.

Blockchain for checking licenses?

Let us imagine how Blockchain could be used for licensing purposes. This is the home turf of Wibu-Systems with our solution that is favored by thousands of publishers (Independent Software Vendors, or ISVs) and millions of users.

Let us now use Blockchain for CodeMeter. Blockchain would be kept in identical copies at all ISVs. Company "A-CAD" could, for instance, see how many licenses company "B-CAD" has created. Even if the data is kept in encrypted form, this would not be a good idea, as every member of the chain would, at the very least, see the number of transactions happening around them and draw their own conclusions. Let us ignore the problem of the sheer amount of data that all ISVs and their users have to keep and keep updating.

The imaginary scenario obviously takes us nowhere. Let us instead imagine a scenario with one Blockchain per ISV. Again, one end user – say,

architect "Tom Dick Associates" – could see how many licenses his competitor – architect "Harry and Partners" – has in use. And again, the sheer amount of data would be prohibitive.

CodeMeter uses one central database kept by the ISVs, who have a legitimate interest in monitoring their licenses. They create licenses for their users and give each user the right (in the form of a cryptographic key) to use them, encrypted by CodeMeter. The keys can only be decrypted and used by their legitimate users. The licenses are also signed to prevent tampering. Using the Blockchain language, we could call these truly miniature mini-blocks. The consensus in this case is simple: "The ISV is always right."

CodeMeter : Blockchain 1:0

Blockchain for protecting software?

For ISVs to enforce their licensing models, the software needs to have a way to check licenses reliably and securely. The best method for doing this is encryption: The application or digital content is encrypted and only users who possess the right license can access the keys needed to decrypt it. If possible, the decrypted software or content should be active in a similarly secure environment, like a dongle, system service, or the cloud. This means that the end user never has direct access to the keys.

This is the exact backbone of CodeMeter, which offers the tools for encrypting software and other contents as standard.

CodeMeter : Blockchain 2:0

Unambiguous identification

Another important aspect of licensing and software protection is the correct identification of the people entitled to use a license. CodeMeter does this with a CmDongle, an account in the CodeMeter Cloud, or a CmActLicense securely bound to the user's computer. The software could only be used if the legitimate user has the right license ready in one of these three containers.

CodeMeter : Blockchain 3:0

License Usage Tracking

One interesting use case revolves around tracking how often a software application or other digital right, e.g. the right to 3D-print a certain product, is used. The ISV should be able to allocate the usage rights and to bill the user for the actual usage.

Assigning such rights is a typical use case for CodeMeter, as we have seen in our scenario above. How does the transaction work in the other direction? Let us return to our imagined CodeMeter Blockchain.

We must consider two key questions:

"How can we be sure that the user indeed logs the transaction into Blockchain?"

Suitable measures need to be put into place that make sure that users can only access their software or protected contents if the usage is logged and billed. With CodeMeter, protection and usage tracking are intrinsically linked in the cryptographic system.

"What happens if the user is offline?"

In this case, the transaction cannot be transmitted and booked in. The ISV now has a choice: Should the software be unavailable in this scenario (risking disgruntled customers), or should the lost revenue simply be accepted? A choice between a rock and a hard place. One workaround would be the keeping of a local Blockchain plus delayed reporting. However, as the local Blockchain is not reinsured by the presence of other blocks elsewhere, the most recent blocks could be removed without the manipulation, becoming visible in the distributed ledger.

CodeMeter offers two options for tracking licenses: A tamper-proof counter built into the license itself can be used to track how often it has been accessed. The counter works even in offline scenarios and would later report back to the ISV. If pre-paid licenses are used, the simple count-down could even work without that report.

The other option would be a log file created and protected by CodeMeter with Blockchain-type methods.

CodeMeter : Blockchain 4:0

Working offline

An essential element of the Blockchain is that its data is always current and duplicated across the members of the ledger. This means the connection is always-online. For many industrial environments, this can be a deal breaker.

CodeMeter offers an opportunity to transfer and to use licenses offline. This can be done by an encrypted update file and dongle. A backchannel is only needed for post-paid models which can again happen by file transfer.

CodeMeter : Blockchain 5:0

Borrowing and transferring licenses

A final use case we need to consider is the ability to lend and borrow licenses. For this purpose, the license must be transferred from a license server to a local device, where it should be ready for use even without standing connection to the license server. After a certain period has expired, the license should revert to the server.

To do so, CodeMeter transfers special mini-blocks: The ISV creates a license with a defined and restricted scope, which is activated at the user by the license server.

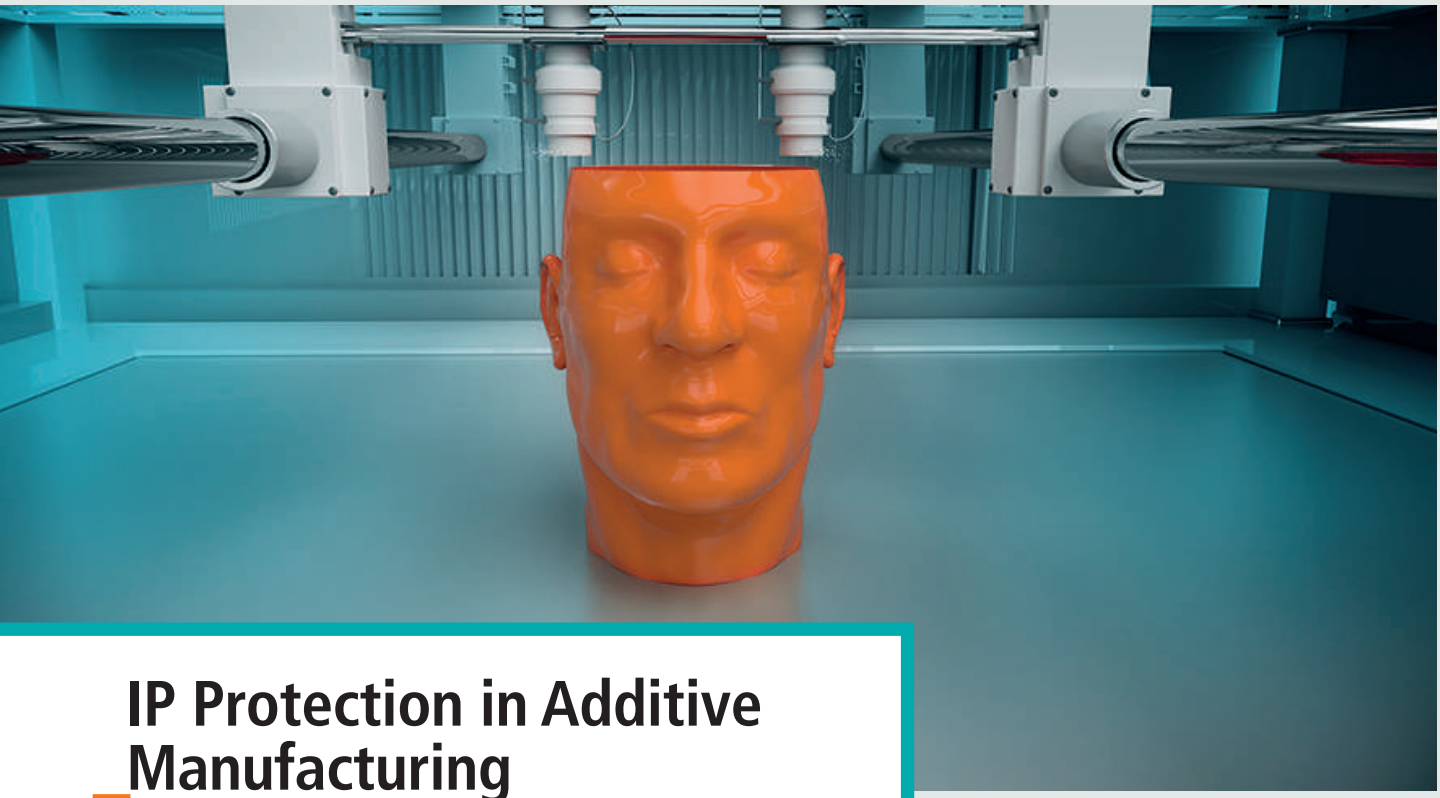
If the ISV allows licenses to be borrowed, CodeMeter has a special protocol for transferring this block to another computer, specifically in a secure CmContainer. The new block is signed by the license server and encrypted for the target container. The borrowing is recorded in the history on the license server, so that the license automatically reverts when the lending period expires. The block is also invalidated when the license is returned.

Compared to Blockchain, the license blocks used by CodeMeter can be cut back and the history deleted after the license has been safely returned. This makes sure that the amount of data used in the process and the performance of the machines involved stays at the optimum level and that the solution scales to meet its demand.

CodeMeter : Blockchain 6:0

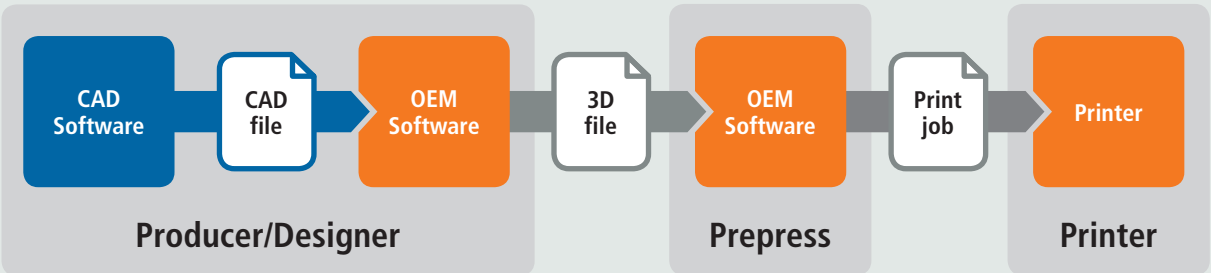
Conclusion

CodeMeter was tailored specifically with the requirements of software protection and licensing in mind. There is one central entity (at the ISV) creating and managing the licenses. The amount of data held by the end user is kept to a minimum. Offline use and offline transfers are explicitly an option. Licensing and protection are inseparable from the cryptographic processes. And some of the familiar methods from Blockchain are used where it makes genuine sense to do so.



IP Protection in Additive Manufacturing

3D printing has morphed over time from a plaything for elite nerds to a viable technology for the future of industry. Companies now have the ability to print components in a wide range of materials on-demand when and where they need them. Numerous international brands have begun to offer devices for the additive manufacturing of e.g. prototypes or spare parts. Even though the printing process itself is deceptively simple, it is an extreme feat of technological innovation, and it remains quite a costly proposition. But as has always been the case with groundbreaking technologies, time will overcome these growing pains and establish 3D printing as a regular part of the industrial experience.

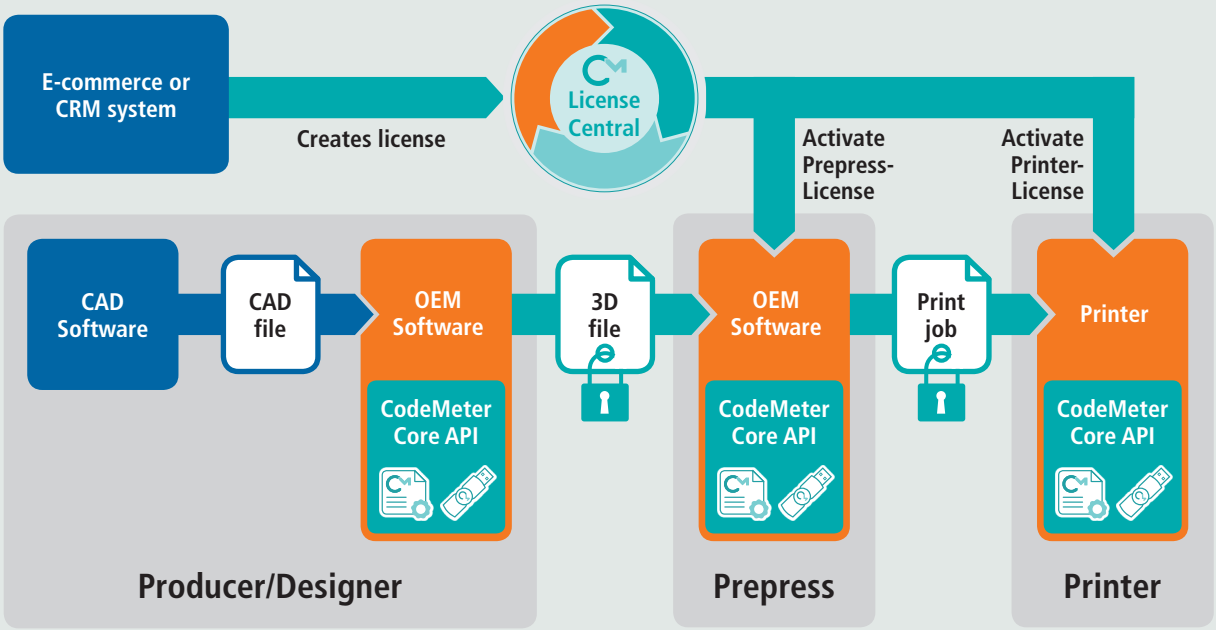


Where are we heading?

The vision pursued by many manufacturing businesses is the ability to produce third-party components right in their own factories in order to respond flexibly to demand in the market, without having to go through their complex supply chains. Ignoring the technological challenge for a minute, this poses another important issue that needs to be considered from the outset: Who is allowed to access the designs – when, where, and how often? A system is needed to protect the underlying data and to monetize the act of printing a third-party design.

Who is involved?

Following the chain from the digital design of an object to the eventual finished product, there are several independent, but interlocked actors who all have a stake in the process. The very first player is the object's designer who has created a 3D blueprint of the piece (e.g. a chair armrest) with a specialized software tool. He would be interested in protecting his blueprints from theft and in having some



means of tracking how many of his pieces are produced, irrespective of when and where in the world this happens.

Before the piece can physically be printed, the data still needs to be processed in a variety of ways. The 3D design data needs to be translated into a layered model, because the actual printers create the pieces additively, i.e. layer upon layer. The material properties (plastic, metal etc.) also need to be considered, as they might change over time or with changing temperatures, which might, in turn, affect the printing process. All of these questions are already covered by dedicated and sophisticated software packages that do the necessary calculations and steer the actual printing process.

These packages do not have to come in one proprietary suite from one software publisher but can be mixed and matched. This makes IP protection through the entire chain an even more complex problem. Finally, the ability to count the number of printed objects must be included in the printer management itself to ensure meaningful controls over the process.

What could a complete IP protection landscape look like?

Even though 3D printing is still in its infancy, IP protection and usage counting can draw on a long history from other areas of application. Encryption, authentication, and licensing have been the bread-and-butter business of Wibu-Systems for more than three decades, and the tried-and-tested CodeMeter technology already includes all the pieces to add state-of-the-art protection and monetization capabilities across the entire digital process chain.

It all begins with the secure encryption of the CAD data (STL files) and the other data processed en route to the actual printer. The CAD software and other applications involved in the other steps need to be able to decrypt the protected data, process it, and encrypt it again before handing it down the line. The cryptographic toolbox of CodeMeter API has the necessary routines on board to integrate encryption and decryption easily and quickly in the various software products and ensure complete IP protection throughout the chain.

At the same time, the security of the entire system rises and falls with the secure storage and allocation of the cryptographic keys needed for encryption and decryption. With CodeMeter, they can be stored securely as licenses in a hardware device (CmDongle), software (CmActLicense), or cloud (CmCloud) container to match the needs of the rights holder. The licenses can also be assigned specific rights, such as restrictions on the feature set or the expiry time. The license used for printing can also be given a quantity counter to track or limit the number of objects the user is allowed to produce.

Managing and safely distributing licenses (rights) is already handled perfectly by CodeMeter License Central. The cloud-based approach allows rights to be allocated to users around the clock and around the globe (and, if need be, offline if the user's system has no direct internet connection). A choice of interfaces is available to allow the integration of back office systems and pave the way for extensive automation.

Now, if a company wants to 3D print a specific component of a third-party supplier for use in their production, they can place an order in the supplier's online store. They would then receive an encrypted file with the design and the rights they need to print the number of objects they have paid for. This gives them enormous flexibility in terms of time and quantities, which can be a substantial financial advantage, especially if the producer and user of the component are based on opposite sides of the world. The original designer of the component can rest assured that his valuable know-how is secure and indeed reaching new target groups and new markets via the novel distribution channel.

Where are we now?

Additive manufacturing is considered a future market – but the future starts today. Many sectors of industry have already realized its potential for small production runs (e.g. in medical technology or consumer and designer goods). This great future potential will, however, be influenced substantially by the fact that the people printing the objects are not necessarily the people who own the rights to them. There are also opportunities for dedicated agencies that can take over 3D printing jobs for other businesses in their vicinity. All of this makes uncompromising IP protection and flexible monetization options an absolute must for the technology.

With CodeMeter, Wibu-Systems is already offering a secure and long-established system that can handle all digital value chains in additive manufacturing.

CodeMeter Cloud Pilot Phase: The Birth of a New Product

The Background

CodeMeter Cloud was first envisaged around four years ago. Wibu-Systems' Corporate Technology department was expanding and looking at how the use of licensing systems was evolving, and they came to an interesting conclusion.

CmDongle and CmActLicenses were still needed, of course, and will continue to be needed into the future, but workforce mobility was mushrooming and the need for real time flexibility becoming apparent. Thus, the concept of CmCloud was first developed.

Of course, Wibu-Systems already provide an ultra-secure dongle-based licensing system that allows for network-based license access, and a slightly less secure software-based license that allows for a degree of portability. But as customers grow and evolve, so do their environments and the requirements of their licensing systems. CmCloud was developed to keep customers fully flexible.

The Pilot Program

At Wibu-Systems, it is very important that we develop solutions that customers actually want. We have never felt that we should simply build products and then explain why a customer should buy them: Rather, we always strive to understand from our customers specifically what they need, why they need it, and what their challenges are, and only then do we develop the software to cater to their requirements.

Most seasoned software professionals have been in projects where features were developed for software and then never used by customers. Sometimes, this happens because customer requirements were not accurately translated into features, sometimes because the features are just assumed: "Well, this is a great idea. Of course people will need that!" I have even known occasions where developers were working on a part of the codebase and decided themselves to add a new feature, because they felt it would be useful in the future.

Essentially, what we are talking about is writing code that does not have a verified use. This is a bad idea for a number of reasons:

- In any modern software environment, all code has an effect on other code and this can introduce instability: Why take the chance?
- New code always costs money to implement, which must ultimately be passed on through higher prices.
- All code needs support, documentation, and maintenance. This simply adds cost to the software and thus should only be undertaken if it is indeed used.

The purpose of the pilot program was to ensure that we at Wibu-Systems understood exactly what customer requirements would be. This was not just from the narrow-minded perspective of what a customer specifically wanted, but rather to understand their environments to ensure that we could anticipate future problems and address them early.

The Pilot Process

We developed a standard process when engaging with a pilot customer. This ensured that we would not miss anything important, but it also helped us refine the process, so that, as new customers came on board, we were able to anticipate any questions they might have and answer them before they were asked.

Setup

As product manager of CmCloud, before I could engage with customers, there were some preliminaries that needed to be addressed. Obviously paperwork needed to be completed, such as the standard Wibu-Systems MNDA (mutual non-disclosure agreement), though this was already in place with existing customers.

	Dongle	Soft License	Cloud License
No hardware costs	No	Yes	Yes
No costs for a cloud services provision	Yes	Yes	No
No shipping costs	No	Yes	Yes
No support costs for computer replacement	Yes	No	Yes
Availability in offline scenarios	Yes	Yes	No
Highest security	Yes	No	Yes
Portability	Yes	No	Yes

Then there was dongle creation, and here is where it gets interesting. Security is of supreme importance to us, and as such all customers are allocated their own FSB (firm security box) that provides the encryption capabilities the customer will need for protecting their software and generating their licenses. This is so complex and secure that Wibu-Systems itself cannot generate licenses for customers' software, but we needed to do so in order to test the CmCloud solution.

Four dongles were created for each pilot customer, each of which replicated only a part of the function of the regular FSBs in use. These were set with different encryption certificates to the customer's original FSBs. In effect, it treated them like new customers and thus could not compromise their original secure setup.

- One to protect the software – sent to the customer
- Two to create licenses for the software – one for the customer, one for Wibu-Systems
- One to create the underlying certificate used by CmCloud – for Wibu-Systems

Of course, it feels odd to create four dongles for a dongleless licensing system, but these four dongles will be replaced in production with the company's own FSB and only used to create licenses and protect the software. It is not needed for end users.

First Contact

The initial contact with the pilot customer was typically performed via an internet meeting and always included me and Dr. Bjoern Grohmann, Head of Corporate Technology at Wibu-Systems. We would explain why CmCloud had been developed and then give a demonstration of its use. Bjoern had written a simple visual application and was able to show how usage could be controlled remotely and in real time through CmCloud.

The easiest way to understand this is typically through the use of license quantity (LQ), which, in this environment, sets the number of times an application can run concurrently before CmCloud prevents startup. With LQ=1, only one instance of a licensed application can run at any particular time, while LQ=3 allows three instances to be used simultaneously. Obviously, other standard Wibu-Systems licensing metrics can be used in a similar manner, e.g. expiration time or license quantity. It is easy to see how licensing models can be built up around this structure.

Subsequent Meetings

I wrote a document explaining how to get up and running with CmCloud and always sent this to the pilot customer. Thus everyone was able to use CmCloud without any problems and as product manager this is particularly important to me: the software should be as easy to use as possible, since any problems that can be ironed out at an early stage.

Initially, we tried to have meetings every fortnight, depending on the availability and wishes of the pilot customer. After a while, this dropped to the occasional email to ensure they were happy and not stuck on anything.

Feedback was encouraged at every step of the way. Positive feedback is always welcome, of course, but constructive criticism is even more important, since this is what we need to improve the software - the whole point of the pilot program!

Feedback

The Good

The following comments are all 100% genuine and accurate, though clearly any identifying information has been removed since security is paramount.

"I really like this. It looks very cool."

"And it looks very good. Actually I can use <one of our major products> as usual...A first performance test looks very good."

"Very happy that it integrates so effortlessly."

"…the concept is really promising … finally a real solution for licensing in virtual environments."

The Bad

During the pilot phase, we discovered areas where CmCloud did not deliver exactly as the customer needed. We diligently logged these requirements and prioritized them, since the whole point in writing software is to provide what customers want.

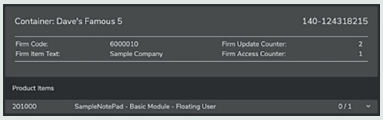
"Proxy support needs improving" was probably the biggest comment we received. With so many different types of networks with so many different implementations of proxies on so many platforms, it is a particularly hard problem to solve; but, because we were able to get this feedback early, we were indeed able to ensure CmCloud worked smoothly in proxy-restricted environments.

Variations of "I'd like some license reporting" were also recorded. It might not sound like it, but this is actually a huge product in itself, since "reporting" can involve a multitude of different nuances. For a while, we had been aware that a reporting element would likely be needed, so it was actually a very positive thing to receive confirmation of this. Thus we have been working on a reporting provision, for example, that the customer can see how many instances of a particular license is in use at any time. More enhancements will follow.

"We need credential management for large customers" was also a popular requirement. This is again a huge topic – one that warrants its own article – but suffice it to say, this will be provided.

The Ugly

Nothing to see here: the software proved itself to be resilient and performant.



Conclusion

I think it is fair to say that the whole pilot program continues to be a huge success (we are still taking new pilot customers on). I have always found it to be of the utmost importance for customers to be able to talk directly to me about the products that I manage, so that I can personally understand the reasons for any pain: in that way, I can get problems ironed out more effectively. From a personal perspective I feel the pilot program was a huge success, because I was able to chat informally to pilot customers and get to know them a little better.

CmCloud is due to be released in December. Thanks to the invaluable feedback I received from our pilot customers, it will be better than it would have been and, for that, I want to thank them all.

News in Brief

CodeMeter Embedded 2.32

The latest version of our static library specifically designed for embedded systems is now supporting MIPS implementations, porting to ARM64 platforms, VxWorks SR 06x0 versions, and BlackBerry QNX 7.0.

BlackBerry | QNX

CodeMeter License Central 3.30

The latest release of CodeMeter License Central includes an expanded license history that captures all actions throughout the license lifecycle, supports the fundamental operations for programming TMR (Triple Mode Redundancy) licenses, and simplifies the creation of update items.

CM License Central

Protecting the environment is a top priority

We are redesigning all of our product packaging and selecting suppliers who are strongly committed to green waste circuits. Our new CmDongles packaging solutions are not just aesthetically pleasing, but also sustainable and more stable to accommodate all our form factors.

Latest news from Japan

Our newest regional team is on fire: They are now active members of both the Edgecross Consortium and the Robot Revolution & Industrial IoT Initiative, which will spark meaningful technical discussions and business opportunities.

EDGECROSS CONSORTIUM rri

Investing in our future

We are building our new head offices and the House of IT Security as part of our commitment to creating attractive workplaces and a physical beacon for IT security. For the formal laying of the cornerstone, Dr. Frank Mentrup and Prof. Dr. Joern Mueller-Quade from the Karlsruhe Institute of Technology were joined by the Mayor of Karlsruhe as our esteemed guests.

Trustworthiness is king

The white paper on "Managing and Assessing Trustworthiness for IIoT in Practice", recently published by the Industrial Internet Consortium and co-authored by our very own Marcellus Buchheit, underlines the importance of understanding and building justified confidence in the key IIoT system characteristics of safety, security, privacy, resilience, and reliability.

CodeMeter in distributed manufacturing

Wibu-Systems is kicking off a collaboration with EOS to build a DRM solution for their 3D printers. Manufacturers of spare parts will then be able to protect their intellectual property, safeguard data confidentiality and integrity, and fully trace and tighten up their digital supply chain.

EOS

R&D is at the core of our DNA

We have just embarked on two new projects, KoMiK, which pushes for innovative communication and co-operation models to support digital collaboration in SMEs, and DigiFAB4KMU, whose goal is to create a digital twin of the buildings we are erecting and protect the digital assets involved. SPS in Nuremberg in November will mark the showdown of ALESSIO, the two-year project we ran with Infineon, the Fraunhofer Institute, Giesecke & Devrient, Siemens, and the Technical University of Munich, Germany that resulted in the development of updatable security solutions for embedded systems in long-lifespan applications.

ALESSIO

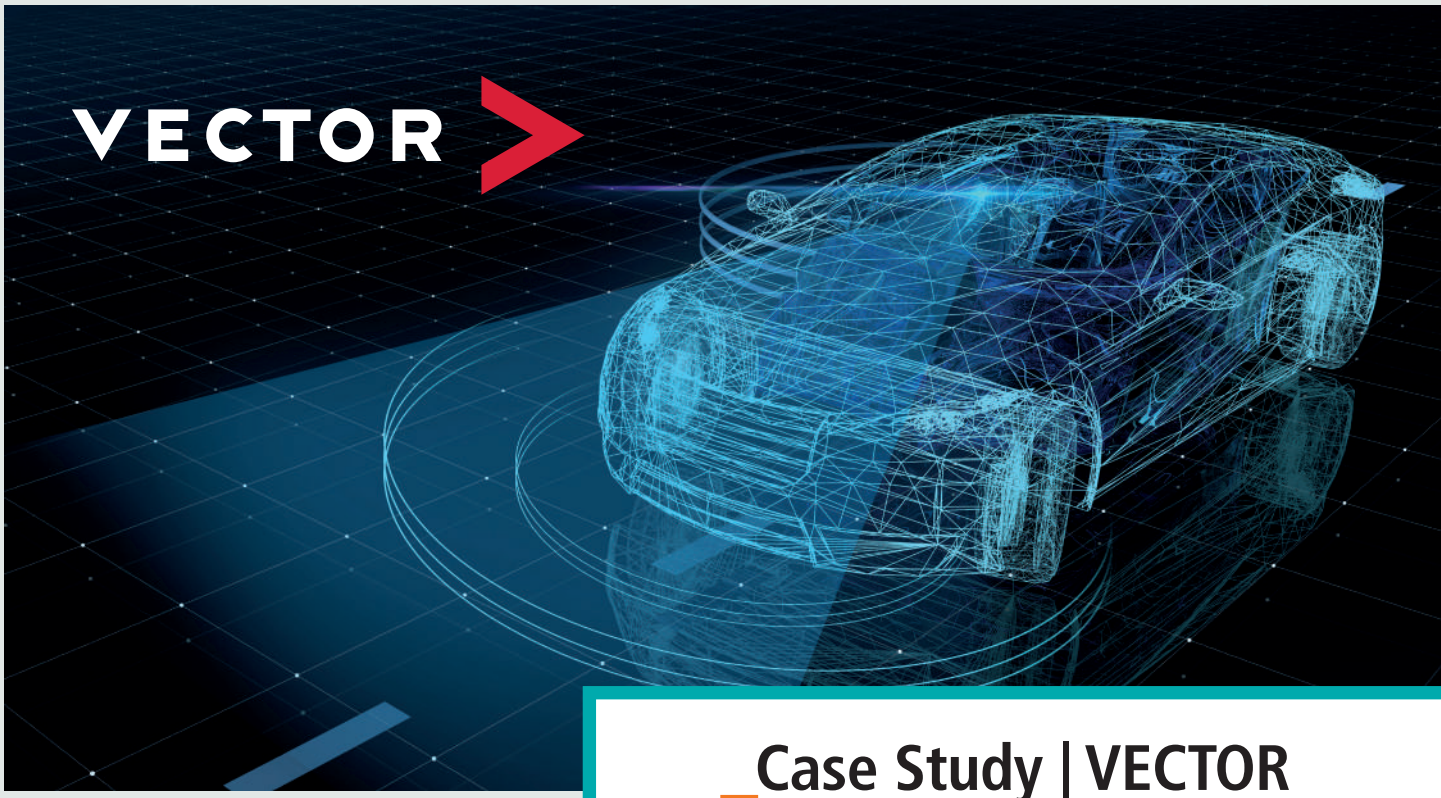
Training the computer scientists of tomorrow

Our partnership with the Baden-Wuerttemberg Cooperative State University (DHBW) enters a new stage: For several years, we have been offering dual courses in computer science. With the start of the new academic year, we are also participating in dual training for software engineers specializing in business informatics. The results are staggering: 80% of graduates from the dual study courses sign permanent employment contracts long before the end of their studies!

DHBW
Duale Hochschule
Baden-Württemberg
Karlsruhe

Siemens selects CodeMeter

Siemens is using the latest generation of CmDongles to realize new licensing models for the SIMIT simulation platform and to protect PROFINET bus analyzers.



Case Study | VECTOR

The Challenge

As a trailblazer for the future of mobility and a leading innovator behind ingenious technologies for electric car charging, safety and security concepts, Advanced Driver-Assistance Systems (ADAS) and autonomous vehicles, and the AUTOSAR Adaptive Platform, Vector's business depends on selling many thousands of licenses every year. Protecting its technology from piracy is a challenge that Vector faces head-on with their Vector Network Interfaces (VNIs) and licenses associated with a combination of Wibu-Systems hardware and software containers. Integrating this system into the production landscape and streamlining its delivery called for a leaner and standardized solution, built right into the company's sales processes.

The Solution

In mid-2018, Vector turned to CodeMeter for a holistic system coupling CmActLicenses, bound with SmartBind® to the target devices for ease of use, and CmDongles for software with more stringent security and mobility needs. A key feature is the use of CmActLicenses, stored in and bound to Vector's unique VNIs.

The system's backbone for license creation, delivery, and management is CodeMeter License Central, fully integrated with Vector's SAP system by Wibu-Systems' dedicated partner, Informatics, an Austrian centre of excellence for SAP. Wibu-Systems and Informatics bring together many years of experience with SAP and CodeMeter to bear on their shared projects.

The Result

With CodeMeter integrated, Vector can manage its licenses centrally from the comfort of CodeMeter License Central dashboard, making for leaner support and sales processes. As the license and entitlement management system is cleverly connected with SAP, licenses can now be created and delivered immediately upon purchase of a product. Users can access, activate, and start enjoying their Vector software without any delay.

The Company

Since 1988, Vector is a premier independent producer of software tools and embedded components for the development of electronic systems and their networking with many different systems from Controller Area Networks (CAN) to Automotive Ethernet. The solutions and products made by Vector pave the way for the future of mobility in the automotive, aerospace, transportation, and control engineering sectors, empowering engineers to master their highly complex work with unrivalled ease of use. Headquartered in Stuttgart, Germany, Vector serves its customers with a workforce of 2,800 people from 26 locations worldwide.

Peter Decker
Product Manager, Vector
"In our collaboration with Wibu-Systems and Informatics, we immediately saw that we are working with a premier team, equipped with just the SAP and CodeMeter know-how we needed. The experience they contributed allowed us to find effective and immediate solutions to our very specific requirements."

Wibu-Systems Training

Wibu-Systems offers custom training to get you off to a running start with CodeMeter software protection and licensing. The training is offered in the form of company courses, typically hosted as in-house classes on your premises. The standard training program includes three days of courses, which can be adjusted to your needs and level of expertise. You can pick and choose the contents you need and shorten the program to 1 or 2 days. Alternatively, you can add a hands-on workshop to allow your participants to try out their own practice cases.

www.wibu.com/tr

Available Courses

CodeMeter Core Features
- CodeMeter at a glance
- Configuring licenses
- The components of CodeMeter Runtime
- Use as a network server

Software Integration for .NET Assemblies with AxProtector .NET and API
- Encrypting .NET assemblies
- Encrypting individual classes and methods
- Integrating Wibu Universal Protection Interface (WUPI)
- Using CodeMeter Core API

Back Office Integration with CodeMeter License Central
- Configuring products
- Creating licenses
- Integrating license activation in applications
- Setting up and configuring license portals

Contact our local representatives for training courses on site.		
German Headquarters	+49 721 931720	info@wibu.com
Belgium / Luxembourg	+32 2 8086739	sales@wibu.be
Canada & USA	+1 425 7756900	info@wibu.us
China	+86 21 55661790	info@wibu.com.cn
France	+33 1 86266129	sales@wibu.fr
Italy	+39 035 0667070	team@wibu.com
Japan	+81 3 43608205	info-jp@wibu.com
Netherlands	+31 74 7501495	sales@wibu-systems.nl
Spain / Portugal	+34 91 1230762	sales@wibu.es
United Kingdom / Ireland	+44 20 31474727	sales@wibu.co.uk

Join Wibu-Systems and its subsidiaries at the following events:

SPS
26-28 November 2019
Nuremberg, Germany

Hannover Messe
20-24 April 2020
Hanover, Germany

CMEF
9-12 April 2020
Shanghai, China

Embedded World
25-27 February 2020
Nuremberg, Germany

T4M
5-7 May 2020
Stuttgart, Germany

Imprint
KEYnote 38
Edition, Fall 2019

Publisher
WIBU-SYSTEMS AG
Rueppurrer Strasse 52–54
76137 Karlsruhe, Germany
Tel. +49 721 93172-0
Fax +49 721 93172-22
info@wibu.com
www.wibu.com

Responsible for the content
Oliver Winzenried

Editors
Stefan Bamberg
Marco Blume
Ruediger Kuegler
David Paine
Daniela Previtali
Wolfgang Voelker
Oliver Winzenried

Design
Eugen Olchin

Print
Stober GmbH, Eggenstein, Germany

Letters are always welcome. We will protect the confidentiality of sources. Third party articles do not necessarily reflect the opinion of the editorial office. Write us at team@wibu.com

Wibu-Systems expressly reserves the right to change its programs or this documentation without prior notice.

Picture credits:
Cover: 123rf.com/47638696
Page 3: istock.com/MicroStockHub
Page 4: istock.com/serts
Page 6: istock.com/matejmo
Page 8: istock.com/robertiez
Page 10: 123rf.com/47638696
Page 12: istock.com/derrrek

All remaining images are copyrighted by their owner

SECURITY
LICENSING
PERFECTION IN PROTECTION

WIBU SYSTEMS