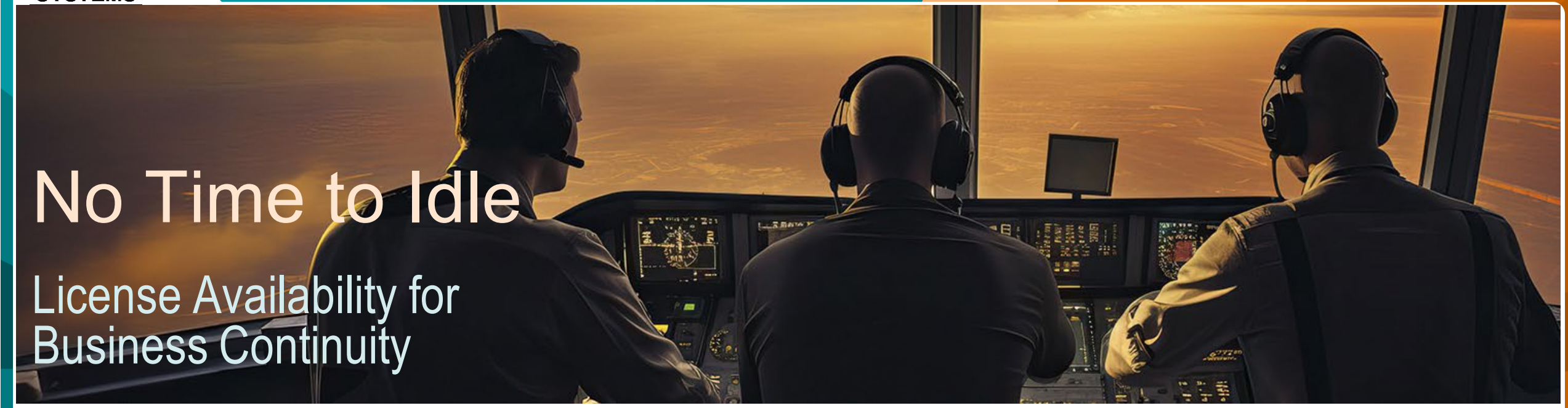
WIBU SYSTEMS



No Time to Idle

License Availability for Business Continuity

Stefan Bamberg

Director Sales & KAM

stefan.bamberg@wibu.com

Wolfgang Völker

Director Product Management

wolfgang.voelker@wibu.com

To access the on-demand replay of this masterclass, please visit

https://www.wibu.com/wibu-systems-webinars/no-time-to-idle-license-availability-for-business-continuity/access.html

Agenda

- Introduction to license management
- Why license availability is key
- Technical approaches for real-life scenarios
- How Triple Mode Redundancy (TMR) works in practice
- Creating and rolling out licenses
- Key takeaways

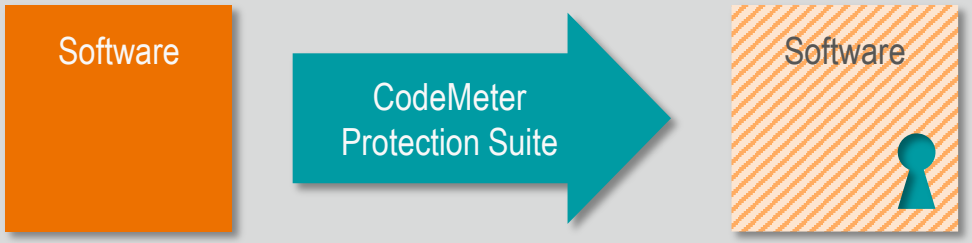
Introduction

The basics of software protection and licensing

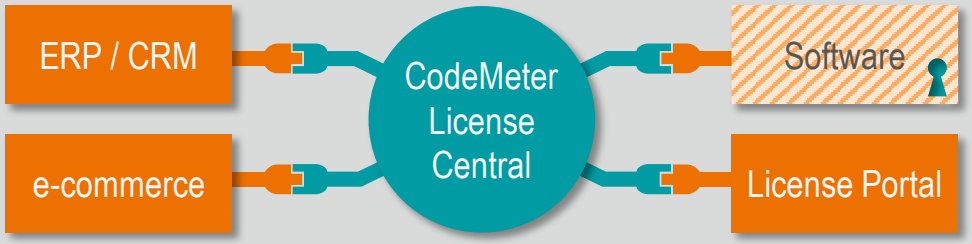
Protecting and Monetizing Software – Quick Overview

Integrate Once

Integration into software

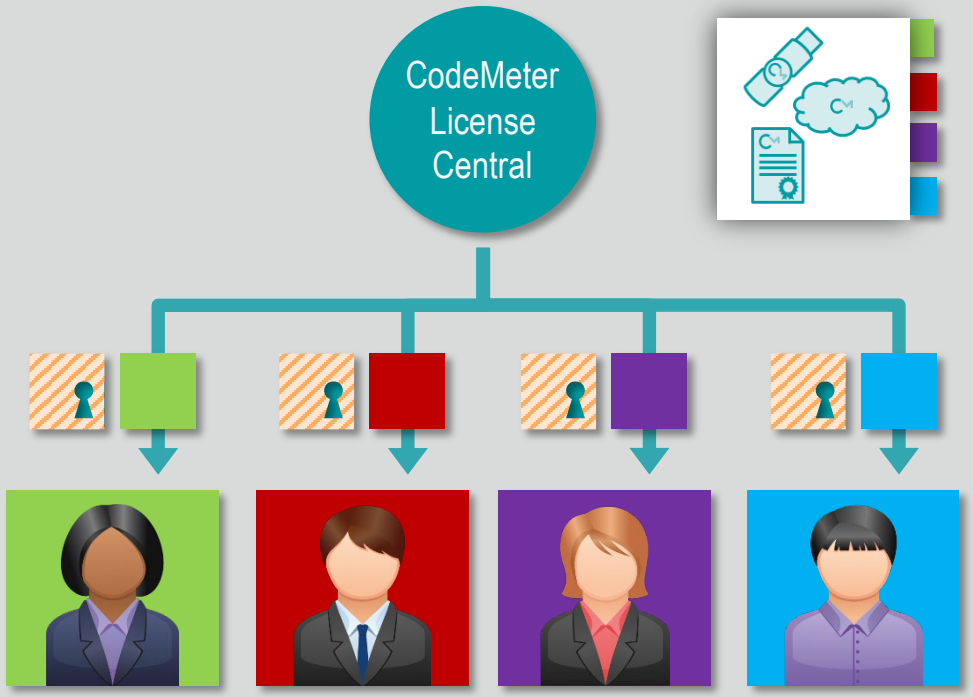


Integration into processes

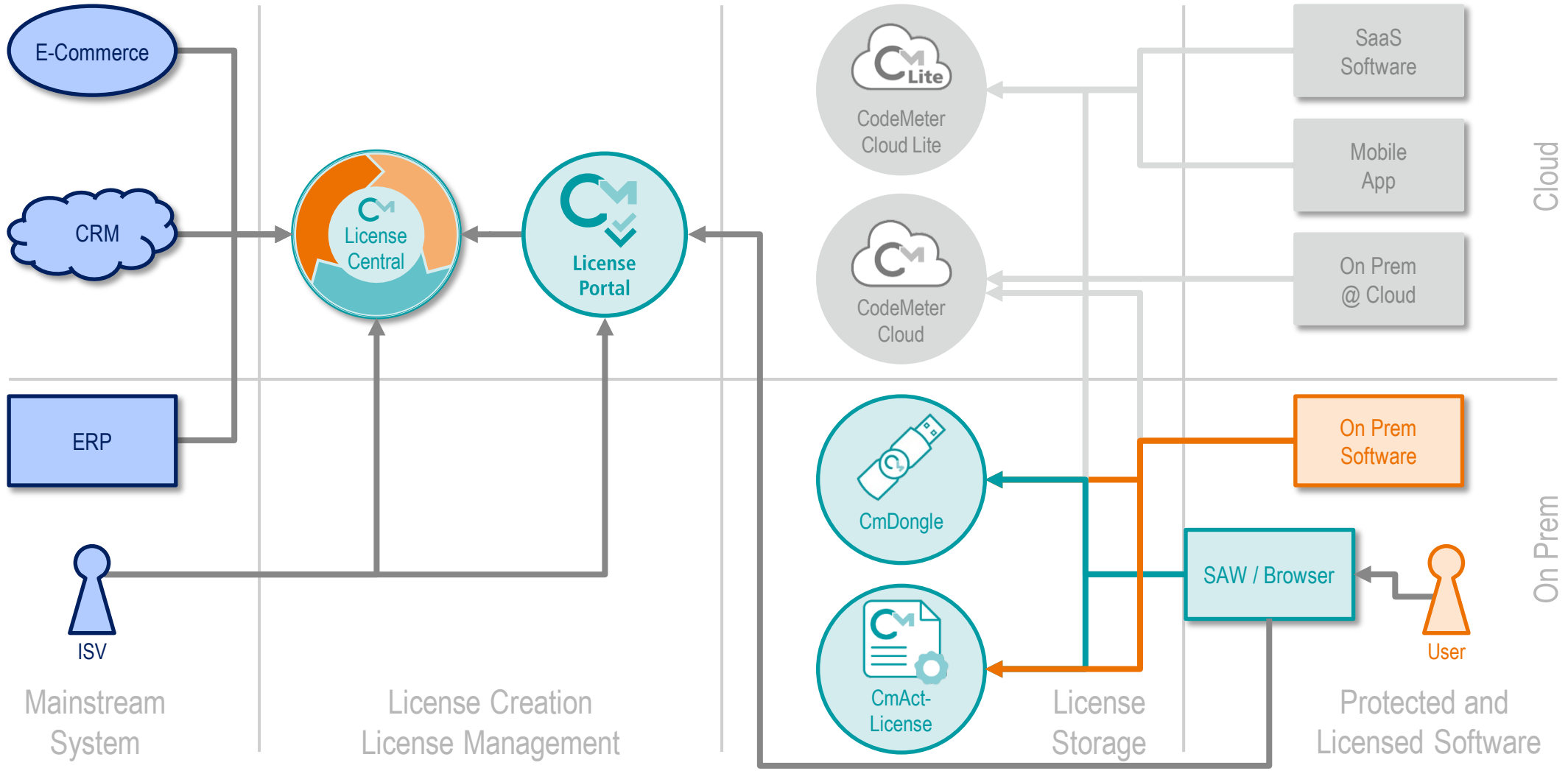


Deliver Many

Delivery to the user



CodeMeter in a Nutshell



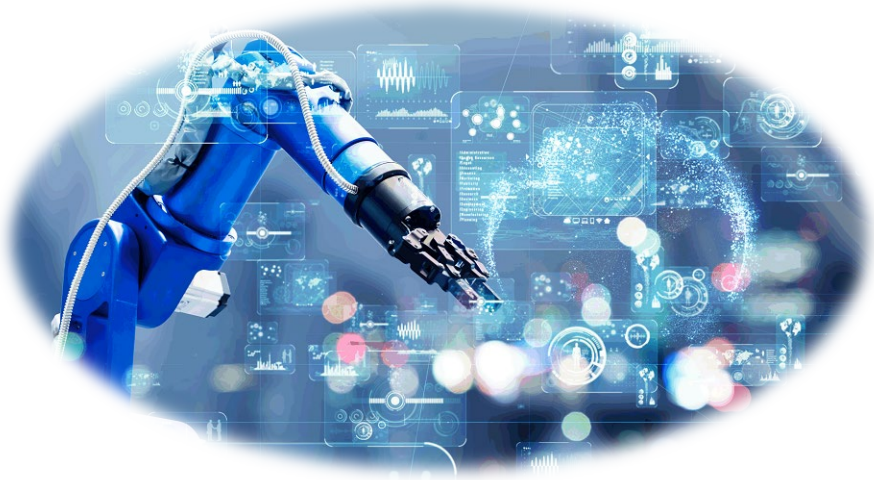
Motivation

Why High Availability is such a Priority?

The Importance of High Availability

- In **production environments or for 7/24 services**
  - In automated production lines and
  - For “round-the-clock” services, e.g. television, radio, petrol stations, ...

  a license failure would lead to significant downtime and associated costs.



The Importance of High Availability (2)

- In the area of **critical applications**
  - Especially for software at the heart of critical infrastructures (e.g. energy, water, healthcare, financial sector, telecommunications, emergency management, etc.), a failure would massively restrict or even endanger public life.

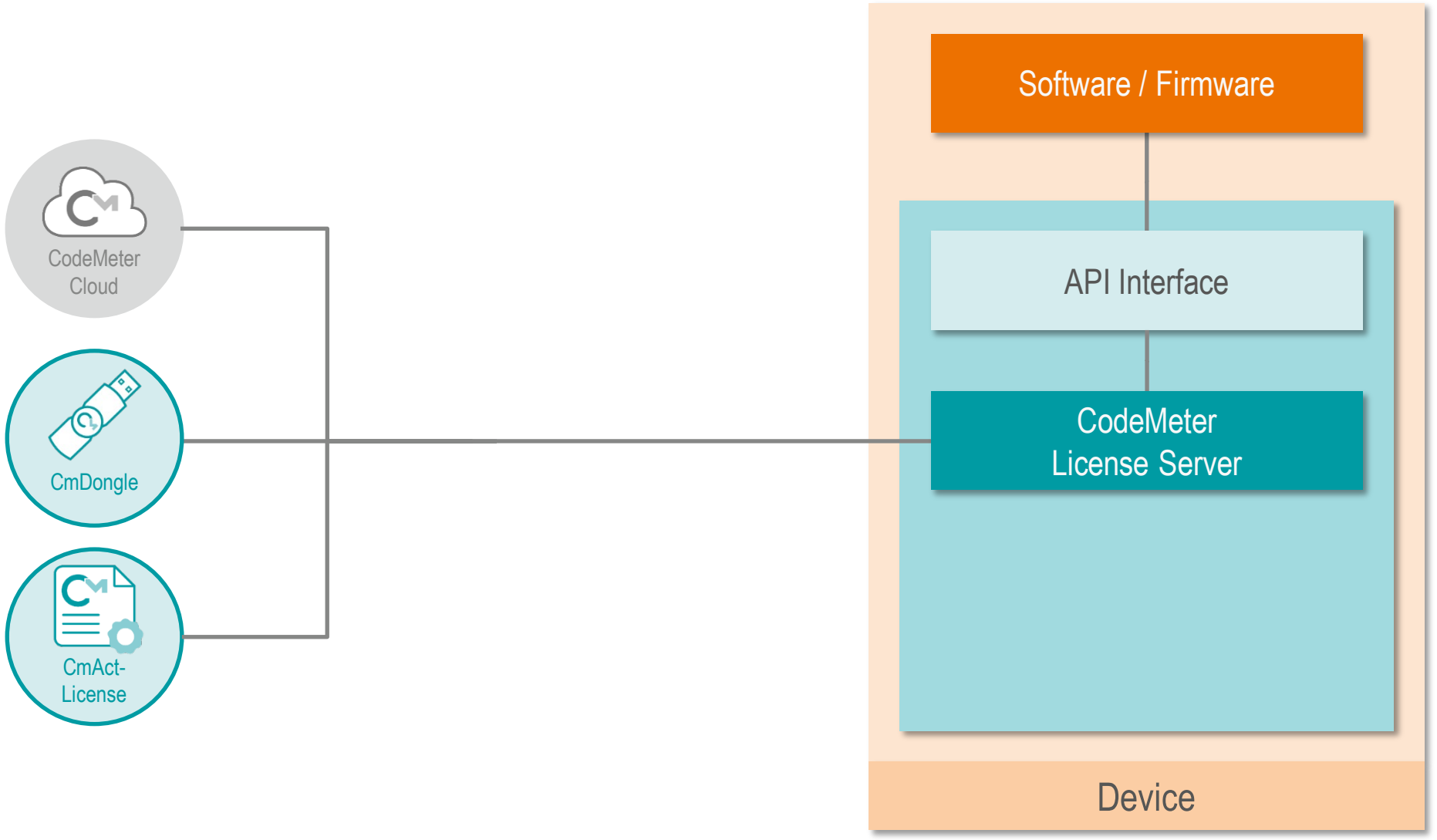
The Importance of High Availability (3)

- With cloud applications
  - More and more software manufacturers are offering Software-as-a-Service in the cloud. This naturally creates a “single point of failure” risk for applications and licenses.

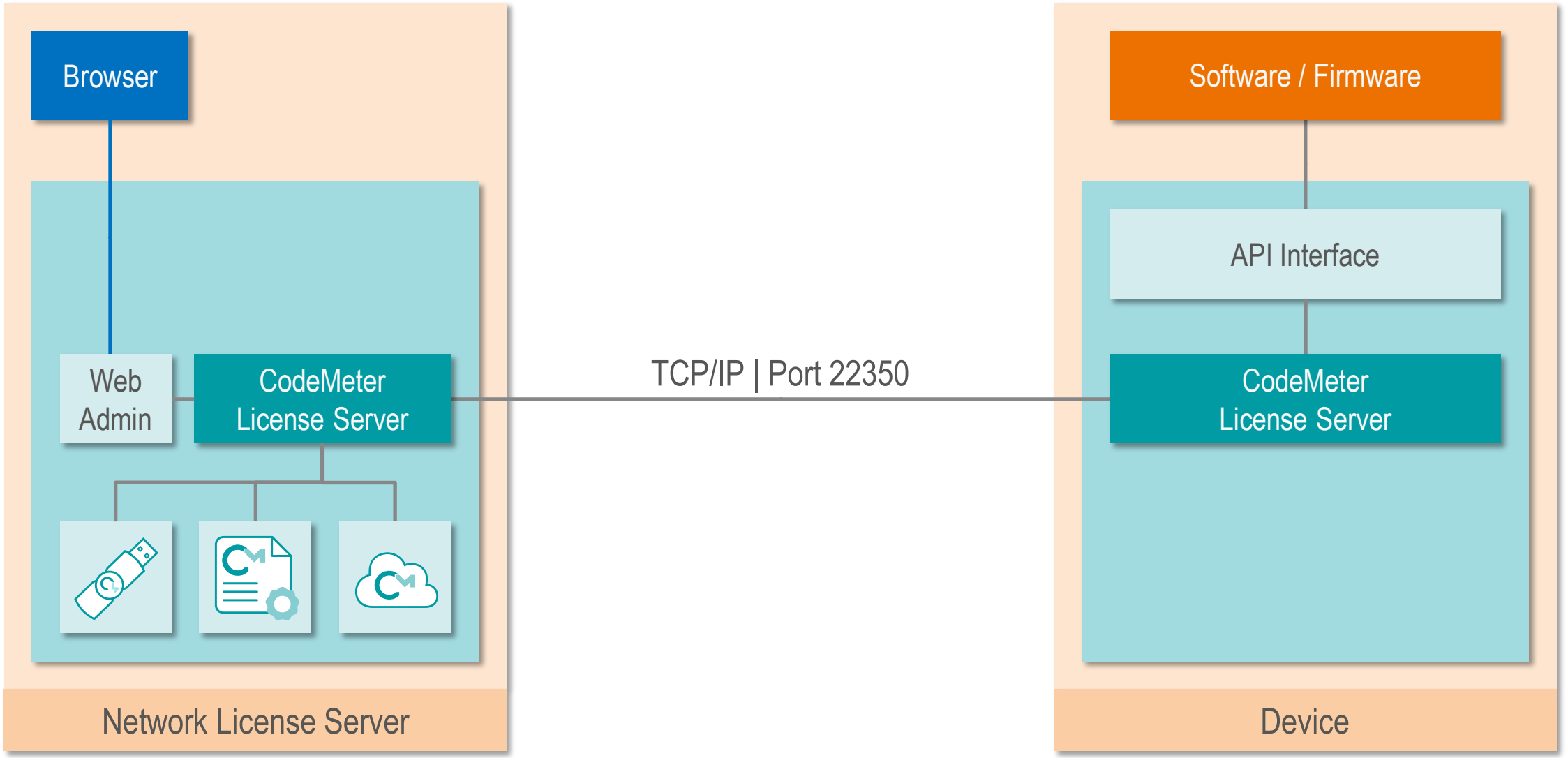
License Availability

- **Requirements of ISVs**
  - Protecting the IP encapsulated in software
    - Know-how is an important asset in software or in the creation of AI models and must thus be protected against reverse engineering and piracy.
      - The solution: encrypting or obfuscating software and data.
  - Monetizing software
    - For many companies, the sale of such software is the existential basis for success. Therefore, it must be ensured that the use of software in the field can be monitored and controlled.
      - Integrating protection and licensing mechanisms in software secures your business.
- **Requirements of ISVs' customers**
  - If the customer is in possession of a license, the software must be able to run reliably with this license so that the customer's business success is always ensured.

Scenario 1: Direct Use of Licenses



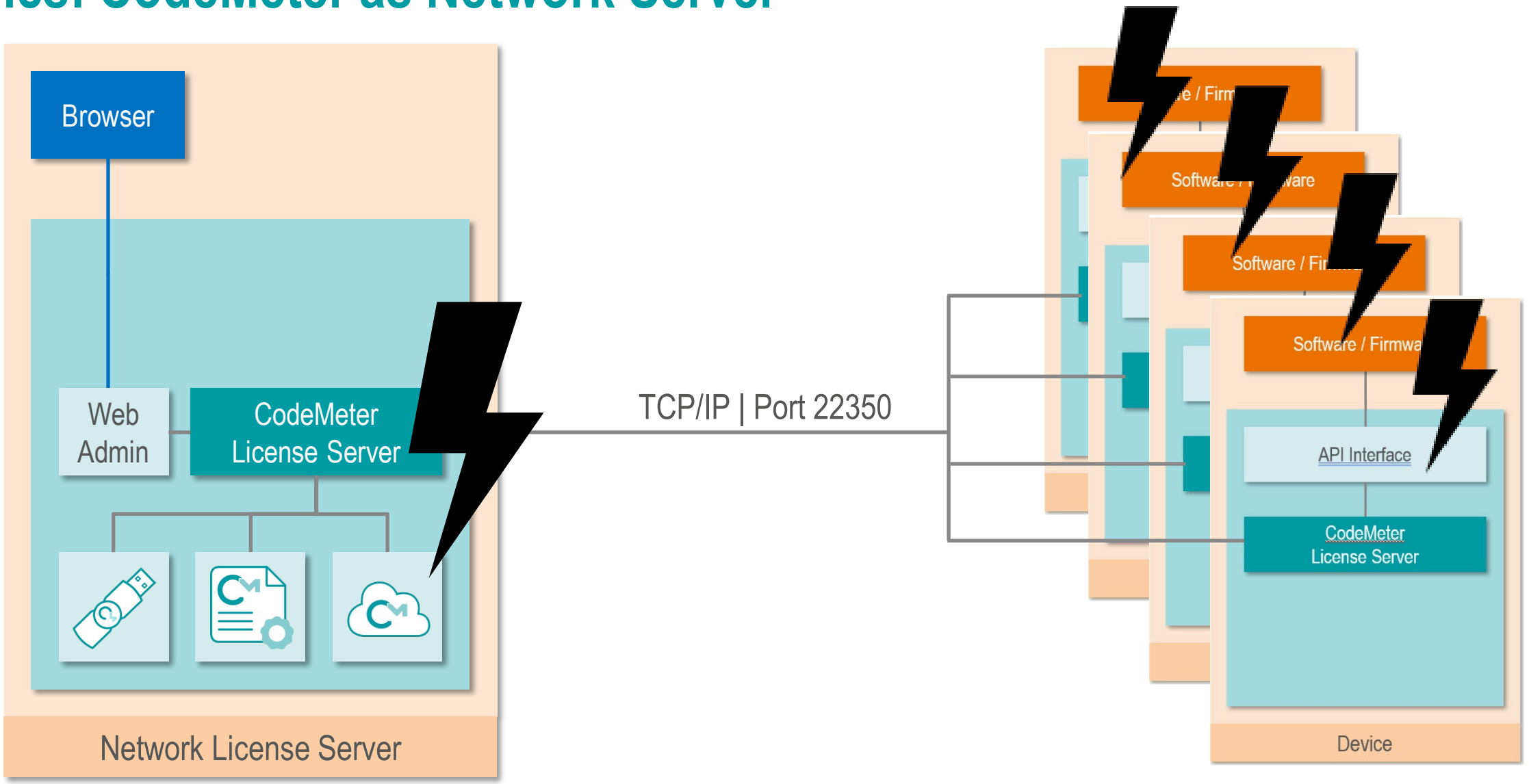
Scenario 2: Licenses via Web Server



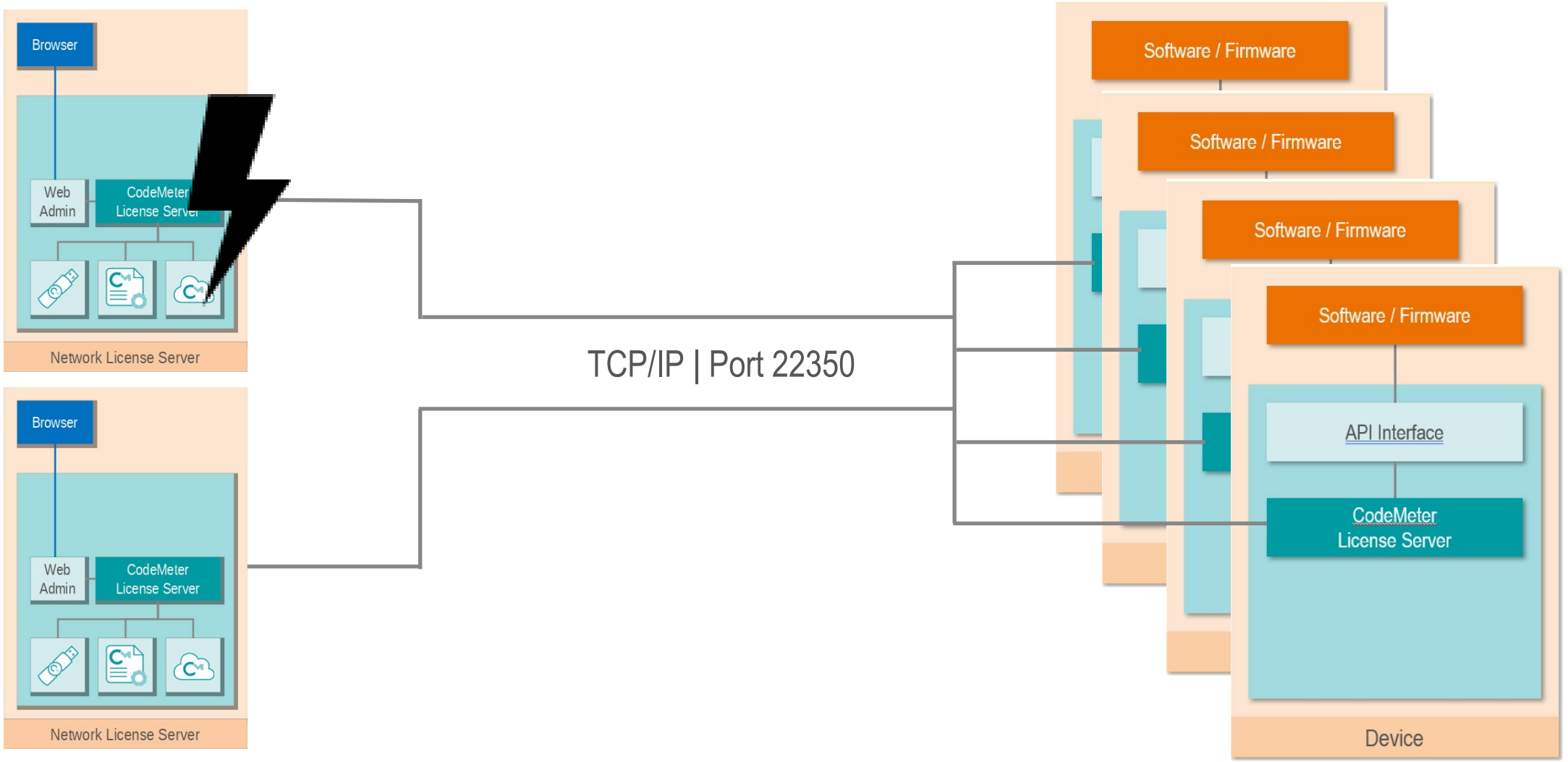
Technical Approaches

High Availability in Practice

Basics: CodeMeter as Network Server



“Simple” Fail-Safe Protection in the Network



Simple Fail-Safe Operation

- **Installation of additional license servers in the network**
  - The ISV's customer sets up another license server in their network
    - **Hot Standby**: The second license server is permanently in operation and is activated automatically or manually if problems occur with the first license server.
    - **Cold Standby**: The second license server is switched off and is started up and made available if problems occur with the first license server.
  - The problem of license misuse: The customer requires additional licenses with the same configuration for the additional license server; therefore, he has twice as many licenses as actually purchased.
    - Potential solutions:
      - Contractual determination of use without any control from the ISV
      - Discount prices for emergency licenses
      - Emergency licenses with a limited duration

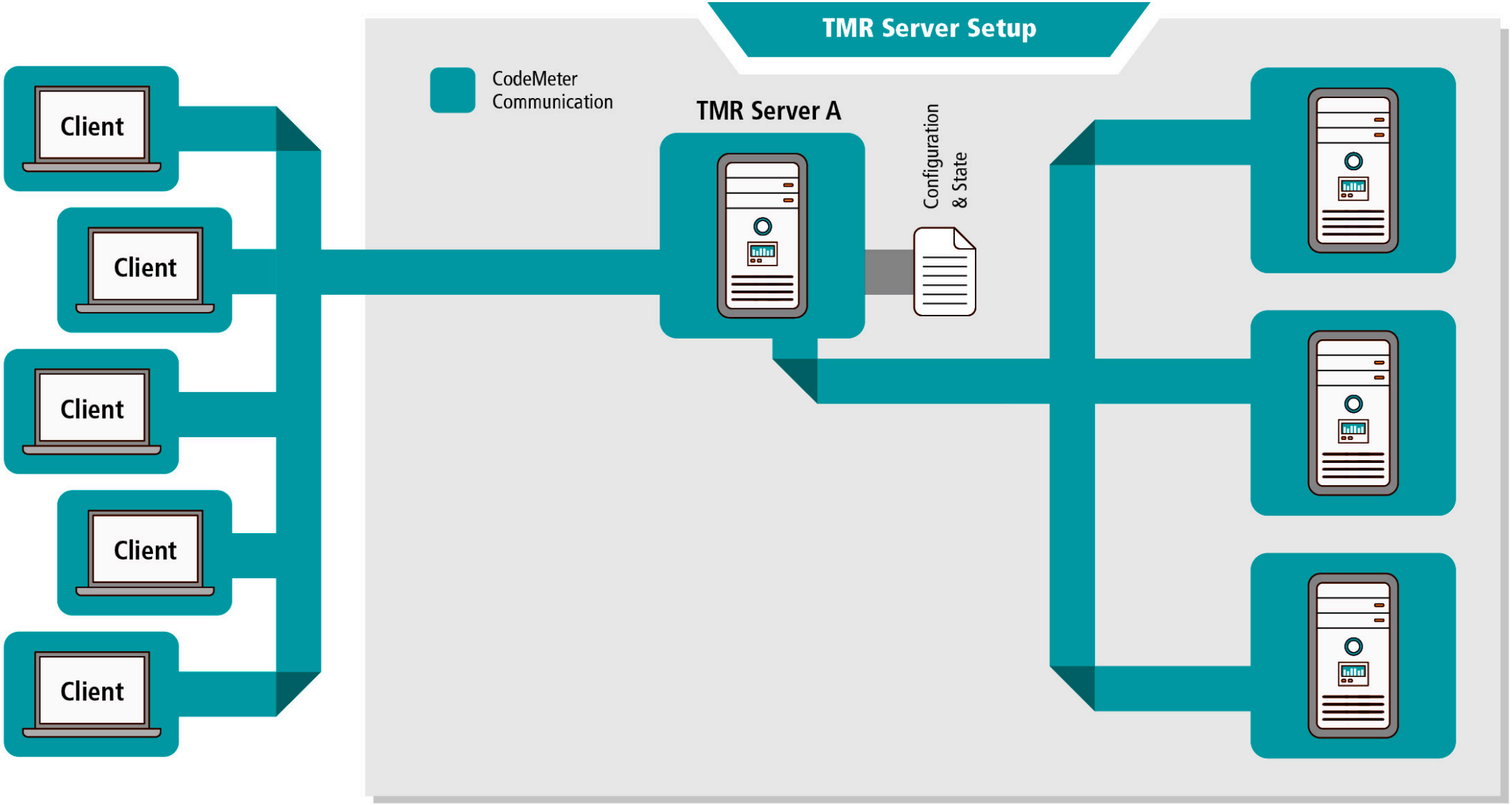
Triple Mode Redundancy (TMR)

Technical Concept

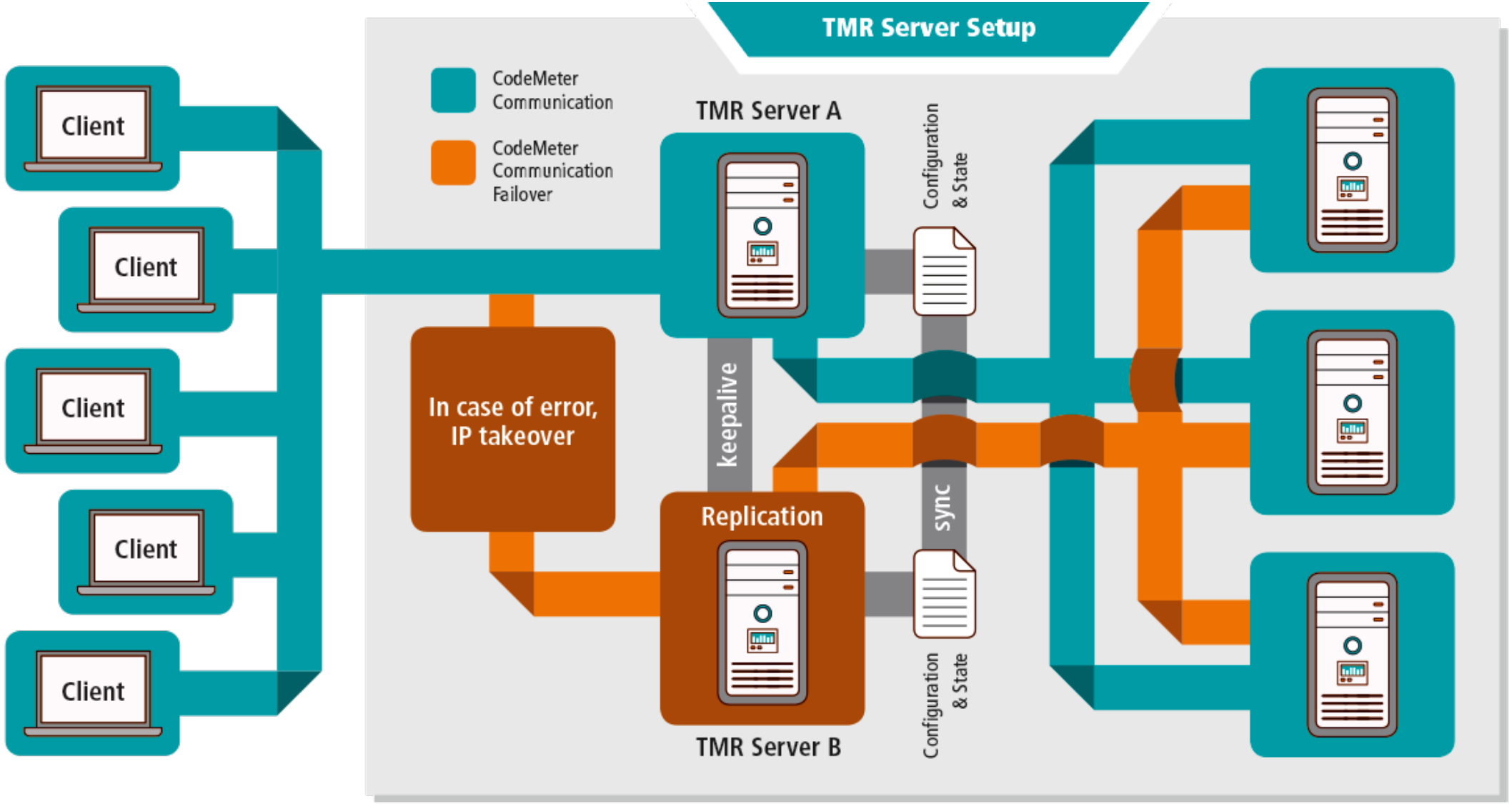
Reliability in the TMR System

- Goal setting:
  - Provisioning the number of licenses purchased by the customer under high availability conditions without the possibility of license misuse.
  - Simple implementation of the concept while retaining the proven process chain during rollout.
- Setting up a TMR system (5 servers):
  - 1 x active TMR server as the interface to the clients who want to use the licenses
  - 1 x hot standby TMR server as backup for the active TMR server with automatic takeover (keepalive daemon)
  - 3 x backend servers providing the licenses.
- The individual servers are each installed on different systems in order to minimize the effects of hardware failures.

TMR network structure



TMR network structure



License Security in a TMR System

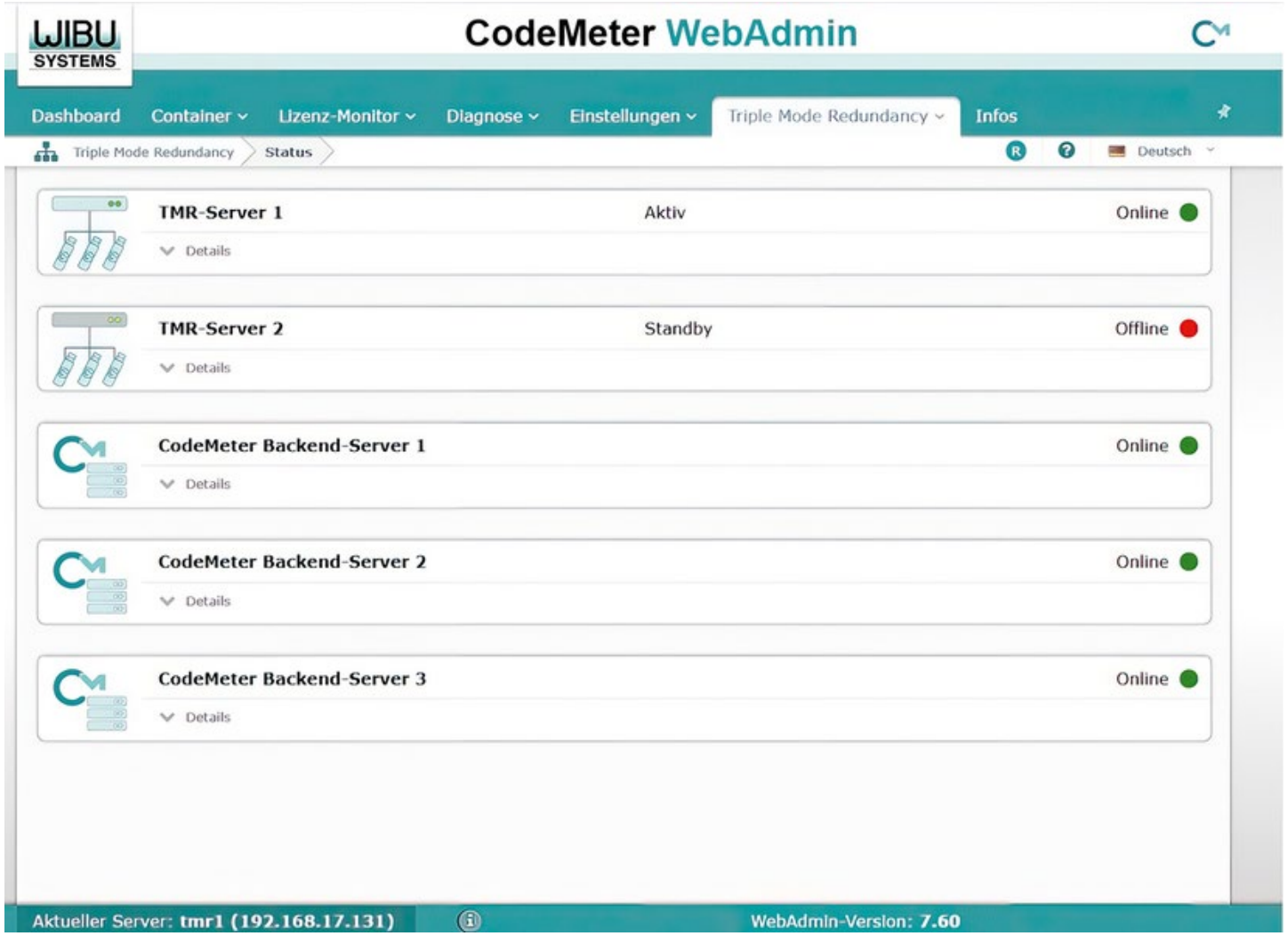
The “2 out of 3” principle:

- The total number of licenses is provided in separate license containers on all three backend systems.
  - These licenses are marked as belonging together via an additional property of the number of licenses (TMR ID).
  - The TMR server uses the TMR ID to identify related licenses and presents them to the clients as one assignable license.
- At least two of the three backend servers and their license containers must be available for the clients to have access to their licenses.
  - Licenses with a set TMR ID cannot be assigned to individual CodeMeter license servers.
    - This rules out any misuse of the license.
  - Maintenance work can be carried out without the need for a maintenance window.

Reliability of a TMR System

Scenario	TMR Server A	TMR Server B	Backend 1	Backend 2	Backend 3	Client Access
Normal operation	Active	Standby	Available	Available	Available	ok
Normal operation	Standby	Active	Available	Available	Available	ok
Maintenance/Failure TMR server (1)	Active	Maintenance / Failure	Available	Available	Available	ok
Maintenance/Failure TMR server (2)	Maintenance / Failure	Active	Available	Available	Available	ok
Maintenance/Failure Backend (1)	Active	Standby	Maintenance / Failure	Available	Available	ok
Maintenance/Failure Backend (2)	Active	Standby	Available	Maintenance / Failure	Available	ok
Maintenance/Failure Backend (3)	Active	Standby	Available	Available	Maintenance / Failure	ok
Failure (1)	Failure	Active	Available	Failure	Available	ok
Failure (2)	Failure	Active	Available	Failure	Failure	Error

License display - WebAdmin on TMR server



Things to know about TMR

- TMR
  - Supports Universal Firm Code (6.xxx.xxx)
  - Provides virtual CmContainers with serial number 131-xxxxxxxxxx
  - Supports CmDongles and CmActLicenses on the backend server
  - No changes to the client required
    - Compatible, for example, with software that was protected by AxProtector in 2017
  - Use of emergency licenses (usage period) not recommended
  - License borrowing not supported

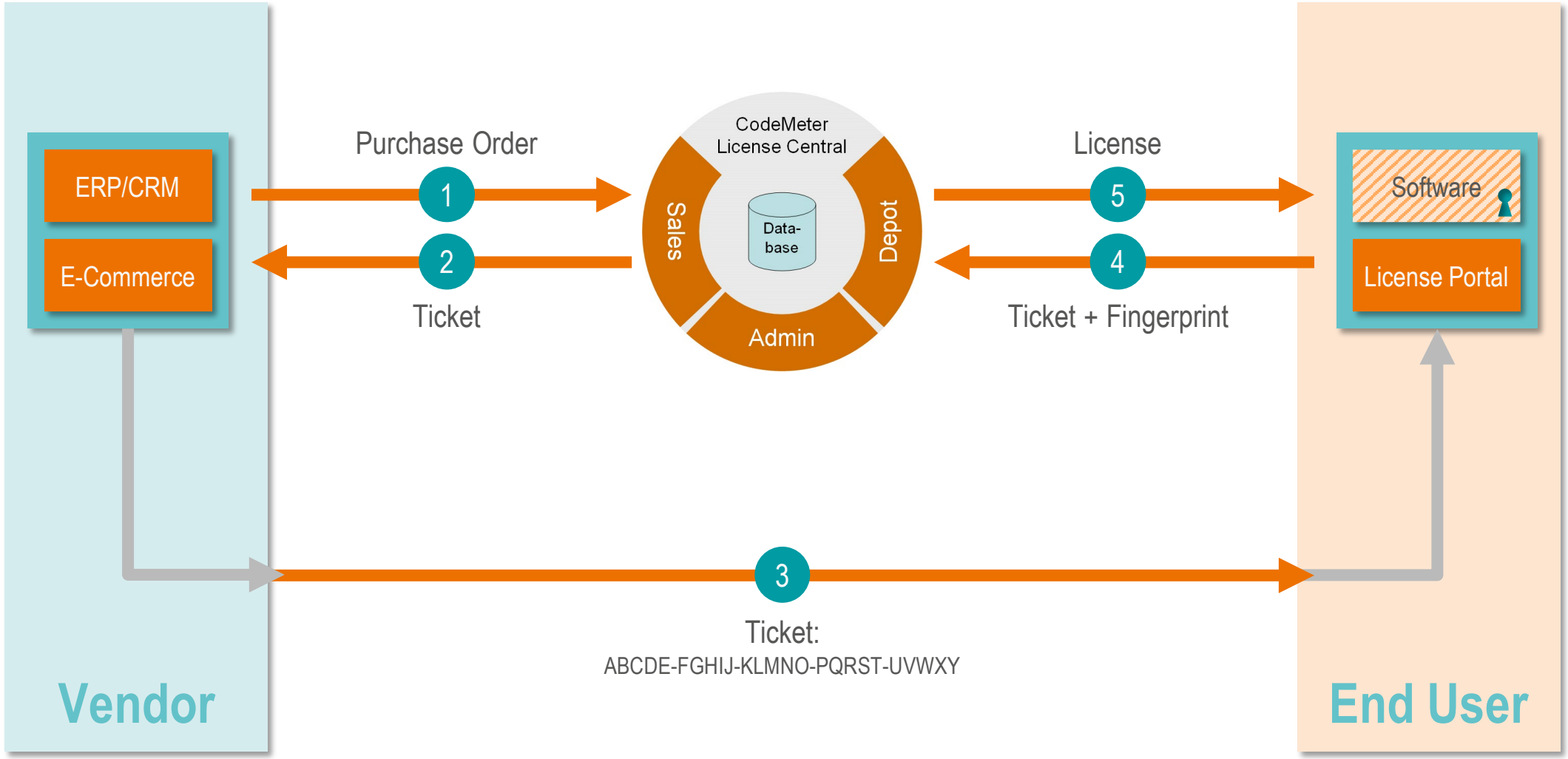
Things to know about TMR (2)

- TMR server
  - Linux system
  - Design based on proven data center technology
    - TMR server pair testing with keepalive
    - TMR server switching using virtual IP
  - Manages licenses / handles
  - Forwards incoming encryption calls to backend servers like a load balancer
  - Processes CodeMeter API calls in parallel
- Backend server
  - Linux or Windows system
  - Configuration as TMR backend server

Creating and rolling out licenses

Using TMR Licenses

Standard Process for Rolling out Licenses



Prerequisites for Licenses in a TMR System

- Activating TMR capability in CodeMeter License Central
- Defining license containers in a TMR system

Global Settings

License Defaults	
Enable TMR	☑

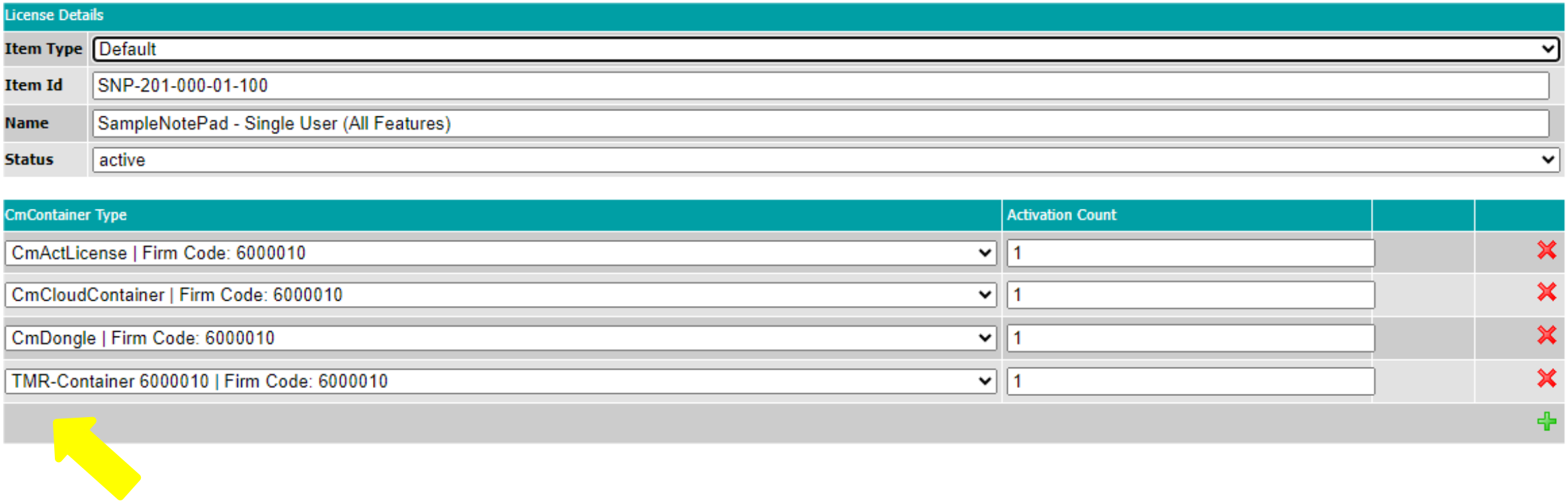
Create CmContainer Type

CmContainer Type	
Name	TMR CmContainer Type
Firm Code	6000010 | Universal Firm Code CmContainer
Usable for	CmActLicense | 2xxx
CmActLicense ID	(is automatically generated)
CmActLicense Name	TMR CmActLicenseName Test
Activation method	Internet
Binding mode	SmartBind
SmartBind tolerance level	Medium
Operating system	☑ All supported Windows versions ☐ Mac OS X ☑ Linux
CmActLicense Options	☑ Allow activation in VM ☐ Allow reimport
TMR Exclusive	☑

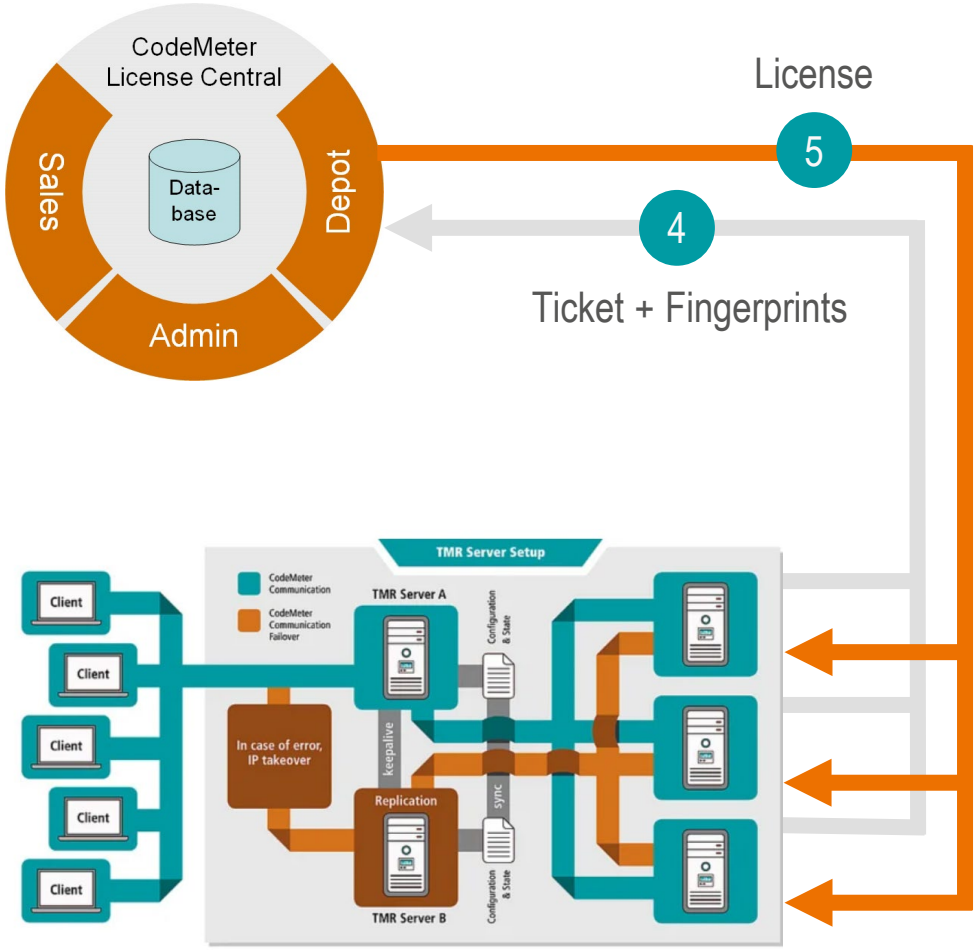
Automatic Restore Settings for TMR	
Initial number of restores	1
One more restore after	1
Maximal number of restores	◉ unlimited ○ value 0

Defining Licenses in a TMR System

- Extending an item definition in CodeMeter License Central:

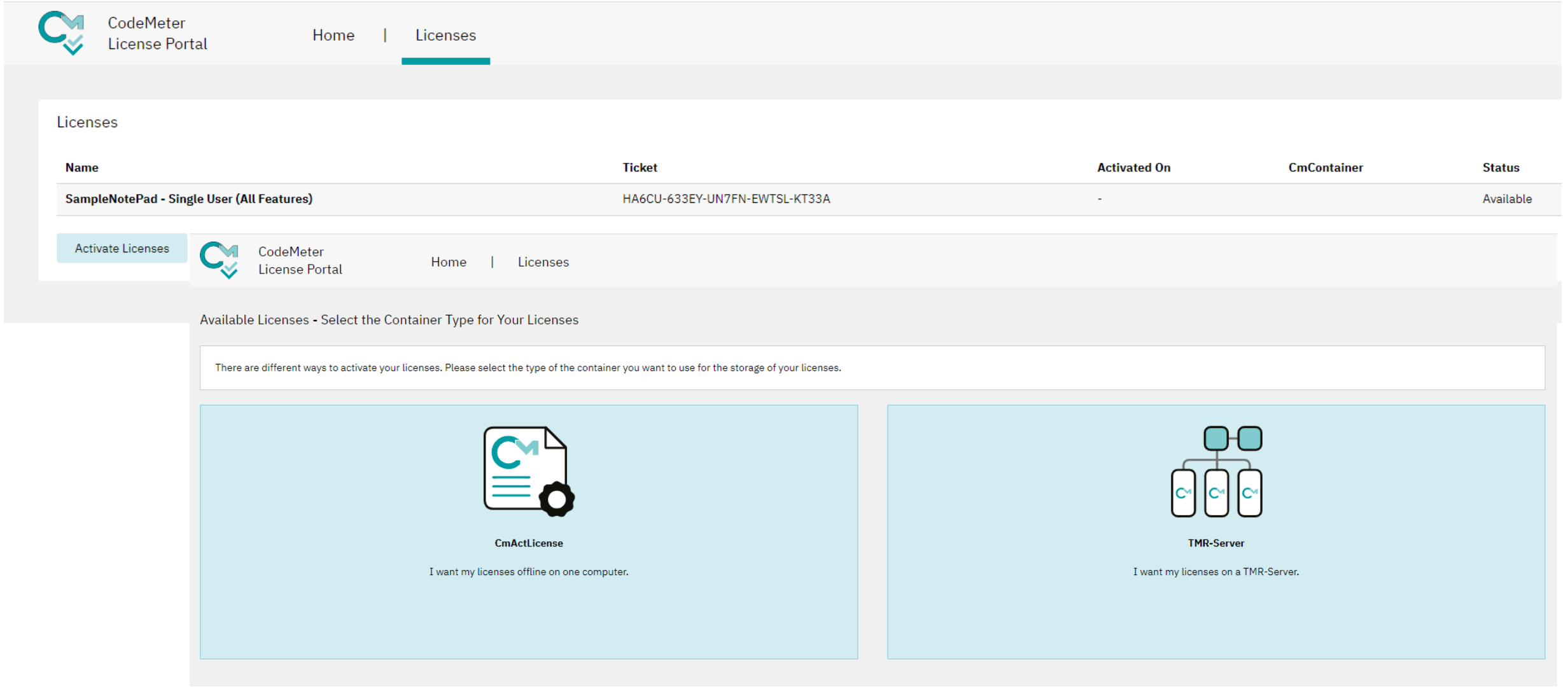


Rolling out and Updating Licenses in a TMR System



- Using the standard processes for license activation:
  - Automatic collection of fingerprints from all three backend systems (step 4).
  - Automatic installation of the licenses on all three backend systems (step 5).
  - Confirming licenses are installed.
  - Available in online and offline environments.

Activation via WebDepot



Installing an Update

- Expiry of an update related to a batch of 20 to 30 licenses
  - License Quantity = LQ
  - TMR ID = #

Process Stage	Backend 1 130-1874829	Backend 2 130-27845973	Backend 3 130-3879234	TMR Status	License in 131-4837572
Before any Update	LQ: 20 / #7	LQ: 20 / #7	LQ: 20 / #7	1:#7 2:#7 3:#7	LQ: 20
Update (Backend) 1	—	LQ: 20 / #7	LQ: 20 / #7	1:- 2:#7 3:#7	LQ: 20
After Update 1	LQ: 30 / #8	LQ: 20 / #7	LQ: 20 / #7	1:(#8) 2:#7 3:#7	LQ: 20
Update (Backend 2)	LQ: 30 / #8	—	LQ: 20 / #7	1:(#8) 2:- 3:#7	LQ: 20
After Update 2	LQ: 30 / #8	LQ: 30 / #8	LQ: 20 / #7	1:#8 2:#8 3:(#7)	**LQ: 30**
Update (Backend 3)	LQ: 30 / #8	LQ: 30 / #8	—	1:#8 2:#8 3:-	LQ: 30
After Update 3	LQ: 30 / #8	LQ: 30 / #8	LQ: 30 / #8	1:#8 2:#8 3:#8	LQ: 30

Key Takeaways

TMR System

Summary

- In the realm of software licensing, high availability is a necessary prerequisite for business continuity in many areas.
- The unexpected unavailability of licenses can cause enormous financial damage and have detrimental consequences.
- Simple high availability can lead to license misuse.
- The Triple Mode Redundancy (TMR) concept ensures high availability and prevents license misuse.
- When using TMR licenses, the entire established workflow for rolling out licenses is preserved.

Type your questions in the chatbox

Thank You!

Let’s keep in touch

Europe: +49-721-931720

USA: +1-425-7756900

China: +86-21-55661790

Japan: +81-45-5659710

https://www.wibu.com

info@wibu.com