

Beraterprofil

Name: Peter Most

Kontaktdaten: PERA Software Solutions GmbH
Ulrich-Nanshaimer-Straße 29
85652 Landsham

Webseite: www.PERA-Software.com
E-Mail: PMost@PERA-Software.com
Telefon: 089/909 692 48
Mobil: 0176/240 254 82



Beruf:

- Senior Softwareentwickler C++, Java SE
- Gesellschafter/Geschäftsführer

Bevorzugter Einsatzort:

- Vor Ort: Großraum München
- 100% Remote: Deutschlandweit

Schwerpunkte:

- C++20, CMake
- Java SE 21, Maven
- C++ Modernisierung
- Systemprogrammierung
- Anwendungsentwicklung
- Plattformübergreifende Programmierung
- Multithreading/Multicore Programmierung
- Netzwerkprogrammierung, Interprozesskommunikation
- Programmiersprachen Interoperabilität

Betriebssysteme:

- Windows
- Linux

Methoden:

- Qualität vor Quantität
- Agile Entwicklung (Scrum)
- Test Driven Development (TDD)
- Objektorientierte Analyse-, Design- und Programmierung (OOA, OOD, OOP)
- Continuous Integration (CI)

Sprachen:

- Deutsch (Muttersprache)
- Englisch (Sehr gute Kenntnisse in Wort und Schrift)

Ausbildung:

- Softwareentwickler (Selbststudium)
- Bauzeichner
- Mittlere Reife

Sonstiges: Ich bin leidenschaftlicher Softwareentwickler mit hohem Qualitätsanspruch.

Stand des Profils: April 2024

C++ Kenntnisse

Erfahrung:	• Experte (1990 - heute)	
-------------------	--------------------------	--

Versionen:	• C++11, C++14, C++17, C++20	
-------------------	------------------------------	--

IDEs:	• Visual Studio 2022	++++
	• Qt Creator	++++

Buildtools:	• CMake	++++
	• Vcpkg	+++
	• Conan	++

Unit Tests:	• GoogleTest	+++
	• Catch2	++

Code Coverage	• OpenCppCoverage	++
----------------------	-------------------	----

Standard Libraries:	• STL	+++
	• Boost	+++

Socket Libraries:	• Boost.Asio	+++
	• BSD Sockets	+++
	• Qt Sockets	+
	• Named Pipes	+

GUI Libraries:	• Qt 5	++
	• wxWidgets	+

Languageinteroperability:	• C-Bindings	++++
	• JNA	+++
	• C++/CLI	+++

Multithreading/Multicore:	• Windows Threads	+++
	• POSIX Threads	+++
	• Parallel Patterns Library (PPL)	++
	• Threading Building Block (TBB)	+

Logging:	• Boost.Log	++
	• NLogC	++

Datenbanken:	• SQL	++
	• SQLite3	++

Java SE Kenntnisse

Erfahrung:	• Sehr gut (2005 - heute)	
-------------------	---------------------------	--

Versionen:	• Java SE 5.0 - 21	
-------------------	--------------------	--

IDEs:	• Eclipse	++++
	• IntelliJ	+++

Buildtools:	• Maven	+++
	• Eclipse	+++
	• Ant	++

Unit Tests:	• JUnit5	+++
	• SikuliX	+++
	• Mockito	++
	• EasyMock	++
	• PowerMock	++

Code Coverage:	• EclEmma	+++
-----------------------	-----------	-----

Standard Libraries:	• Java Platform SE (JDK)	+++
	• Google Guava	+++
	• Apache Commons	+++

Socket Libraries:	• java.nio	++
--------------------------	------------	----

GUI Libraries:	• Swing	++
	• JavaFX	++

Languageinteroperability:	• JNA	+++
	• ComfyJ	++

Multithreading/Multicore:	• Threads	+++
----------------------------------	-----------	-----

Logging:	• Log4j	+++
	• LogBack	++

Datenbanken:	• SQL	++
	• Derby, Oracle	++
	• JPA	++

HTML:	• rendersnake	+++
	• jsoup	++
	• Apache FreeMarker	++

XML:	• JAXP	+
-------------	--------	---

Sonstige Kenntnisse

Versionsverwaltungen:

- Git +++
- Jazz ++

Issuetracker:

- GitHub ++
- Bugzilla ++
- Jazz ++

Nebenprojekte

Projekt:

"C++ AidKit"

Beschreibung:

Design und Entwicklung der "C++ AidKit", eine Cross-Platform C++/Qt/Boost Library mit wiederverwendbaren Klassen und Funktionen.

Technologien:

- C++20, CMake, Vcpkg
- Qt 5, Boost, STL, Google Test
- Qt Creator, Visual Studio 2022
- Git, Visual Studio Code
- Linux / Windows

Homepage:

<https://github.com/petermost/CppAidKit>

Projekt:

"FritzBoxCallMonitor"

Beschreibung:

Design und Entwicklung von "FritzBoxCallMonitor", eine Cross-Platform C++/Qt Applikation zum Anzeigen von eingehenden Anrufen auf der FritzBox.

Technologien:

- C++17, CMake, Vcpkg
- Qt 5, Qt Sockets, Qt XML, Google Test, C++ AidKit
- Qt Creator, Visual Studio 2019, Sourcetrail
- Git, Visual Studio Code
- Linux / Windows

Homepage:

<https://github.com/petermost/FritzBoxCallMonitor>

Projekt:

"DeleteOldFiles"

Beschreibung:

Design und Entwicklung von "DeleteOldFiles", eine Cross-Platform C++/Qt Applikation zum rekursiven Löschen von alten Dateien.

Technologien:

- C++11, CMake
- Qt 5, C++ AidKit
- Qt Creator
- Git
- Linux / Windows

Homepage:

<https://github.com/petermost/DeleteOldFiles>

Kundenprojekte

Projekt:

"Sourcetrail"

Beschreibung:

Weiterentwicklung von "Sourcetrail", eine Cross-Platform C++/Qt/Java Applikation zum Visualisieren von Klassen- bzw. Funktionshierarchien.

Kunde:

PERA Software Solutions

Aufgaben:

- Einbau der Indexierung von 'User defined conversion operators'.
- Einbau der Unterstützung für Visual Studio 2017/2019/2022.
- Einbau der Unterstützung für Java Standard 16 - 21.
- Einbau der Unterstützung für Catch2 V2/V3.
- Einbau der Unterstützung für libclang-16/17.
- Umstellen der C++ Abhängigkeiten auf Vcpkg Pakete.
- Umstellen der Java Abhängigkeiten auf Maven Pakete.
- Vereinfachen der CMake Build-Skripte.

Laufzeit:

12.2023 – heute

Technologien:

- C++20, CMake, Vcpkg
- Java SE 21, Maven
- Boost, Qt 5, Clang (Libclang/LLVM), JNI, SQLite, TinyXml
- Catch2 (V2/V3)
- Qt Creator, Visual Studio 2022, Eclipse
- Git, Visual Studio Code
- Linux / Windows

Homepage:

<https://github.com/petermost/Sourcetrail>

Projekt:

"GUARDUS" (Engine)

Beschreibung:

Modernisierung der "Guardus-Engine", eine C++ Software zur Analyse und Steuerung in der Produktion.

Kunde:

Carl Zeiss MES Solutions

Aufgaben:

- Umstellen des Buildsystems von MSBuild auf CMake.
- Ersetzen von veralteten C++98 Sprachkonstrukten mit C++17.
- Restrukturieren der Libraries zur einfacheren Wiederverwendung.
- Umstellen auf typensichere STL Standard Container.
- Mergen von 'Copy & Paste' Sourcdateien.
- Beheben von Speicherlecks mit Smart-Pointer.
- Sichern der Qualität durch Einführen von Google Test.
- Überprüfen der Testabdeckung mit Code-Coverage.
- Visualisieren der UnitTest Ergebnisse in Visual Studio.

Laufzeit:

09.2020 – 12.2023

Technologien:

- C++17, CMake
 - Boost, STL
 - Google Test, OpenCppCoverage
 - Git, Azure DevOps, Visual Studio Code
 - Visual Studio 2019/2022, Sourcetrail
 - Windows 10
-

Kundenprojekte

Projekt: "GUARDUS" (GuiTests)	Beschreibung: Design und Entwicklung der "GuiTests", ein Java Testframework zum Ausführen von GUI-Tests.
Kunde: Carl Zeiss MES Solutions	Aufgaben: <ul style="list-style-type: none">• Entwickeln eines Frameworks basierend auf SikuliX.• Implementieren von Tests anhand existierender Testszenarien.• Erstellen eines einzelnen "Executable-Jar" zum Vereinfachen des Deployments.• Ausführen der Tests mit der JUnit5 Platform Launcher API.
Laufzeit: 09.2020 – 12.2023	Technologien: <ul style="list-style-type: none">• Java SE 21, Maven• SikuliX, JUnit5, PicoCLI, Log4j2, JNA, ClassGraph, JDBC• Apache Shade Plugin, Apache Commons• Git, Azure DevOps, Visual Studio Code• Eclipse, IntelliJ, Sourcetrail• Windows 10
Projekt: "Common Sequencer"	Beschreibung: Weiterentwicklung des "ExecutionEnvironments", ein C++ Programm zur Automatisierung von Produktionstests.
Kunde: Rohde & Schwarz	Aufgaben: <ul style="list-style-type: none">• Verwenden des C++ .NET Core Hosting zum Laden und Starten von C# Assemblies.• Generieren von C# Code mit T4 Templates.
Laufzeit: 10.2019 – 05.2020	Technologien: <ul style="list-style-type: none">• C++17, CMake, Conan• Boost, Google Test/Mock• C# 7.0, NuGet• .NET Core 2.2, xUnit• Visual Studio 2017, Sourcetrail• Git, Jira• Windows 10, Linux

Kundenprojekte

Projekt:

"Server Based Testing"
(Data Acquisition Service)

Beschreibung:

Design und Entwicklung des "Data Acquisition Service", ein C++ Dienst zum Empfangen und Weiterleiten von Binärdateien.

Kunde:

Rohde & Schwarz

Aufgaben:

- Empfangen und parsen von JSON-basierten "JobTickets".
- Extrahieren des Dateipfades und cachen der Binärdaten.
- Versenden der gecachten Binärdaten mit ZeroMQ.

Laufzeit:

04.2019 – 09.2019

Technologien:

- C++17, CMake
 - Boost, Google Test/Mock, RabbitMQ, AMQP-CPP, ZeroMQ, libcurl
 - Qt Creator, Sourcetrail
 - Git, Jira, Docker
 - Linux
-

Projekt:

"QuickStep"
(TestConsoleRunner)

Beschreibung:

Design und Entwicklung des "TestConsoleRunners", ein C# Programm zum Ausführen von Google- bzw. NUnit-Tests.

Kunde:

Rohde & Schwarz

Aufgaben:

- Erkennen ob ein Google-Test oder ein NUnit-Test auszuführen ist.
- Ausführen der Tests mit der NUnit-API.
- Erstellen einer XML-Ergebnisdatei zur weiteren Verarbeitung in Jenkins.

Laufzeit:

03.2019

Technologien:

- C# 7.0, .NET Core 2.2, NUnit
 - Visual Studio 2017
 - Git, Jira, Jenkins
 - Windows 10, Linux
-

Projekt:

"QuickStep"
(Linuxportierung)

Beschreibung:

Analyse des Portierungsaufwands nach Linux für eine Windows C++ Applikation.

Kunde:

Rohde & Schwarz

Aufgaben:

- Erfassen der Windows API Aufrufe.
- Abschätzen des Aufwands um die Windows API Aufruf zu portieren.

Laufzeit:

09.2018 – 02.2019

Technologien:

- C++14, CMake
 - Boost, Google Test/Mock
 - Visual Studio 2017, Qt Creator, Sourcetrail
 - Git, ClearCase, Jira
 - Windows 10, Linux
-

Kundenprojekte

Projekt: "TeDiSPlus"	Beschreibung: Weiterentwicklung von "TeDiSPlus", eine Java EE Anwendung zur Disponierung von Technikern.
Kunde: BWI	Aufgaben: <ul style="list-style-type: none">• Erweitern bzw. erstellen von Unit-Tests.• Erstellen von SQL-Abfragen mit JPA.• Empfangen bzw. Senden der JSON Frontend-Daten.
Laufzeit: 04.2017 – 08.2018	Technologien: <ul style="list-style-type: none">• Java 8, Java EE 7, Maven• JUnit 4/5, Mockito• GlassFish (Payara), Derby- bzw. Oracle-Datenbank• Eclipse, IntelliJ, NetBeans, Sourcetrail• Mercurial, Jira• Windows 7

Projekt: "CMWmars" (PCAP Dekodierung)	Beschreibung: Design und Entwicklung der "PCAP-Dekodierung", ein C++/C# Programm zum Empfangen und Dekodieren von PCAP-Daten.
Kunde: Rohde & Schwarz	Aufgaben: <ul style="list-style-type: none">• Dekodieren der PCAP-Daten mit TShark (Wireshark).• Starten des TShark Programms mit QProcess.• Senden der PCAP Daten mit Pipes/Channels.• Empfangen der PDML Daten mit Pipes/Channels.• Parsen der PDML Daten mit QXmlStreamReader.
Laufzeit: 01.2016 – 08.2016	Technologien: <ul style="list-style-type: none">• C++11, C++/CLI, Qt 5, Qt Core, Qt XML, Boost, STL• C# 4.0, NUnit• Visual Studio 2013• Jazz

Kundenprojekte

Projekt: "CMWmars" (LoggingServer)	Beschreibung: Design und Entwicklung des "LoggingServers", ein C++ Programm zum Empfangen und Weiterleiten von Socketdaten an eine C#/WPF Applikation.
Kunde: Rohde & Schwarz	Aufgaben: <ul style="list-style-type: none">• Erstellen einer DLL mit einer Java bzw. C# kompatiblen API.• Asynchrone Socketkommunikation mit Boost.Asio und Boost.Thread.• Weiterleiten der Daten im TLV/Boost.Serialization Format.• Interne Entkopplung/Kommunikation mit Signal/Slots.• Marshallen der C++ Daten nach C#.
Laufzeit: 01.2011 – 12.2015	Technologien: <ul style="list-style-type: none">• C++11, C++/CLI (Bridge), Boost.Asio, Boost.Thread, Boost.Serialization, Boost.Signals2• C# 4.0, NLog, NUnit, Moq, P/Invoke, Mixed Mode Assembly• Java SE 7, JNA• Visual Studio 2010, 2013, Eclipse Mars• Jazz, ClearCase

Projekt: "CMWmars" (LegacyFileImporter)	Beschreibung: Design und Entwicklung des "LegacyFileImporters", ein C++ Programm zum Importieren einer Binärdatei in eine SQLite Datenbank.
Kunde: Rohde & Schwarz	Aufgaben: <ul style="list-style-type: none">• Einlesen einer binären Logdatei.• Speichern der Daten in einer SQLite-Datenbank.• Nutzen mehrerer CPU-Cores mit der Parallel Patterns Library (PPL).• Kommunikation der PPL-Tasks mit einer LockFree-Queue.
Laufzeit: 01.2011 – 12.2015	Technologien: <ul style="list-style-type: none">• C++11, C++/CLI, PPL, SQLite, Boost.Variant• C# 4.0, NLog, NUnit, Moq• VTune• Visual Studio 2010, 2013• Jazz, ClearCase

Projekt: "CMWmars"	Beschreibung: Design und Entwicklung von "CMWmars", eine C#/WPF-Applikation zur Visualisierung von Mobilfunktestdaten. Weiterentwicklung von automatisierten GUI-Tests.
Kunde: Rohde & Schwarz	Technologien: <ul style="list-style-type: none">• C# 4.0, System.Data.SQLite, WPF, NLog, NUnit, Moq, SQLite• Java SE 8, SikuliX• Visual Studio 2010, 2013, Eclipse• Jazz, ClearCase

Alte Projekte

Projekte:

- "HardwareService"
- "LoggingServer"
- "MCT-Tools"

Kunde:

Rohde & Schwarz

Laufzeit:

02.2007 – 12.2010

Beschreibung:

- Design und Entwicklung eines Java Wrappers um eine COM-Schnittstelle.
- Design und Entwicklung eines Java LoggingServers zum Aufzeichnen von Socketdaten in eine proprietäre Logdatei.
- Weiterentwicklung einer Java/Swing Applikation zur Visualisierung, Verwaltung und Ausführung von Mobilfunktests.

Technologien:

- Java SE 5, Launch4j, Ant
 - Swing, CORBA, Log4j, FEST, ComfyJ, NIO
 - JUnit 4, PowerMock, EasyMock
 - Jython, Python
 - Eclipse
 - Bugzilla, ClearQuest, ClearCase
 - Windows
-

Projekt:

"FIA"

Kunde:

Vossloh IT

Laufzeit:

05.2006 – 01.2007

Beschreibung:

Weiterentwicklung der "FIA" Prozesse und Einführung von C++ Unittests.

Technologien:

- C, C++, make, Boost.Jam
 - SQL, PostgreSQL, Boost.Test
 - Java SE, Ant
 - Log4j, JUnit, HTTP-Unit, HTML-Unit, XML-Unit, Apache Tomcat
 - KDevelop, gVim
 - Eclipse
 - Windows, Linux
-

Projekt:

"Auslandsauskunft"

Kunde:

SoftMethod

Laufzeit:

12.2005 – 04.2006

Beschreibung:

Design und Entwicklung eines Java/Swing Prototypen für die WSDL-Serviceschnittstelle.

Technologien:

- Java SE, Ant
 - Swing, Log4j, TableLayout, E115, SOAP, ASN.1, Apache Tomcat, Apache Axis, WSDL
 - Eclipse
 - Sun Solaris
-

Alte Projekte

Projekte:

- "Mediation Device"
- "Remote Player"
- "LIRC"

Kunde:

Siemens

Beschreibung:

- Design und Entwicklung des "Mediation Device", ein C++ Programm zum Aufzeichnen und Weiterleiten von IP-Daten.
- Design und Entwicklung des "Remote Player", eine C++/MFC Applikation zur Ansteuerung eines Java Media-Players.
- Design und Weiterentwicklung des "LIRC", eine C++/MFC Applikation zum Extrahieren von gesprächsbegleitenden Meta-Daten.

Laufzeit:

06.2001 – 11.2005

Technologien:

- C++, SCons
 - MFC, wxWidgets
 - STL, Boost, BSD Sockets, Pipes, POSIX Threads, PCRE Library
 - TLV, XML, ASN.1, Reguläre Ausdrücke
 - "Global Windows Hook", COM
 - KDevelop, GDB
 - Windows, Linux (SuSE, FreeBSD)
-

Projekt:

"VR-Networld"

Beschreibung:

Erstellung eines Gutachtens zur Wartbarkeit einer C++ Codebasis.

Kunde:

E & M Software Solutions

Technologien:

- Visual C++ 6.0

Laufzeit:

04.2001 – 05.2001

Projekt:

"Clicktionary,
Clevertrainer"

Beschreibung:

Design und Weiterentwicklung des "Clicktionary" bzw. "Clevertrainer", eine C++/MFC Applikation zum Übersetzen von Wörtern und Redewendungen.

Kunde:

Cleverlearn

Technologien:

- Visual C++ 6.0, MFC

Laufzeit:

04.2000 – 03.2001

Projekt:

"RCC"

Beschreibung:

Design und Entwicklung des "RCC", eine C++/MFC Applikation zum Überwachen und Aufzeichnen von Telefongesprächen.

Kunde:

Siemens

Technologien:

- Visual C++ 5.0, MFC, Named Pipes, Sockets, Threads

Laufzeit:

02.1998 – 03.2000

Alte Projekte

Projekt:

"RAIS GUI"

Beschreibung:

Weiterentwicklung der "RAIS GUI", eine C++/StarView Applikation zur Verwaltung von Rückversicherungsgeschäften.

Kunde:

Softlab

Technologien:

- Visual C++ 5.0, StarView

Laufzeit:

07.1997 – 01.1998

Projekt:

"Playout-Center"

Beschreibung:

Design und Entwicklung eines C++ Kommandozeilenprogramms zum Ansteuern eines Archivroboters.

Kunde:

BETA-Research

Technologien:

- C++, DBTools++

Laufzeit:

02.1997 – 06.1997

Projekte:

- "BodyCare"
- "Card-DLL"
- "Lieferschein"
- "UNIKA"
- "EC-Kartenverwaltung"
- "Konvertierungen"

Beschreibung:

- Design und Entwicklung von "BodyCare", eine C++/MFC Applikation zum Visualisieren von Patientendaten die von Mikroprozessorkarten gelesen werden.
- Design und Entwicklung von "Card-DLL", eine C++ Library zur Kommunikation mit einem Kartenterminal.
- Design und Entwicklung von "Lieferschein", eine C++/TurboVision Applikation zum Ausfüllen und Drucken von Lieferscheinen.
- Design und Entwicklung von "UNIKA", eine Embedded C Applikation zum Anzeigen von Krankenkassenkartendaten.
- Design und Entwicklung von "EC-Kartenverwaltung", eine Clipper Applikation zur EC-Kartenverwaltung.
- Design und Entwicklung von C/C++ Kommandozeilentools zur Konvertierung von Personalisierungsdaten.

Kunde:

Oldenbourg Datensysteme

Technologien:

- C/C++, Clipper 5.0 (DBase-Kompiler)
 - MFC, TurboVision
 - Serielle Schnittstelle (RS-232), ASN.1, T1-Protokoll (ISO 7816-3/4)
-