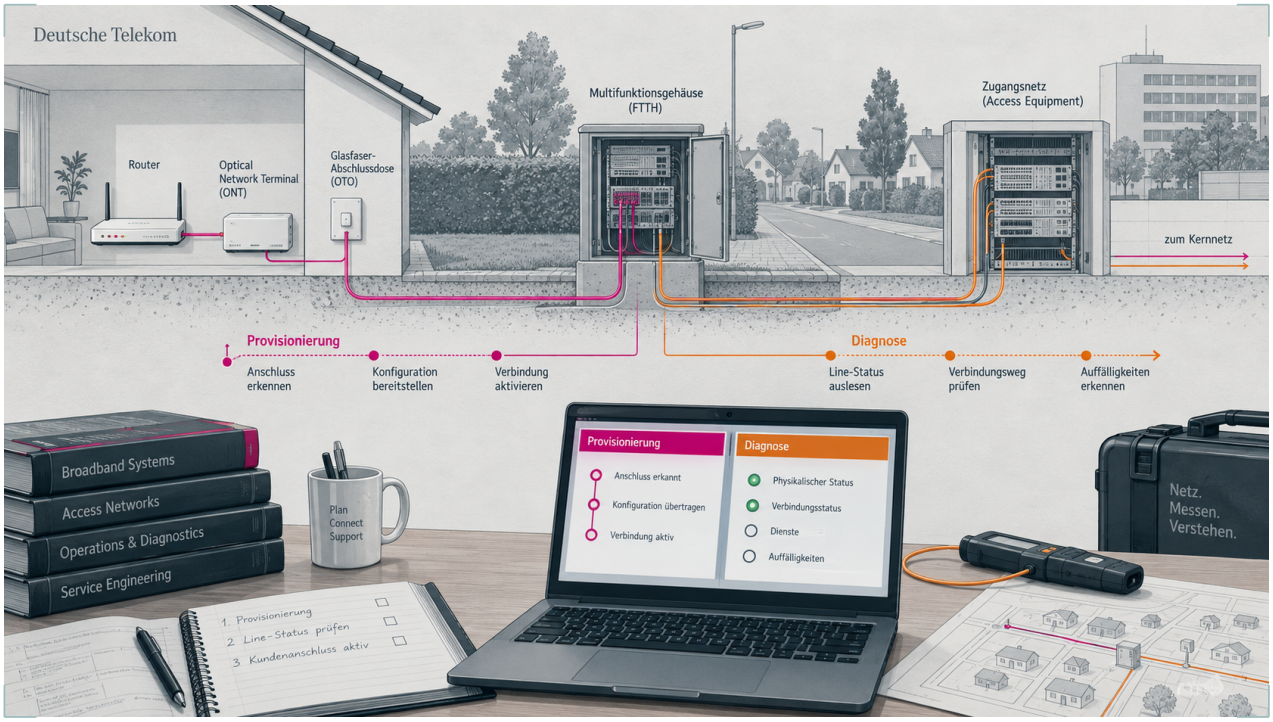
ENGINEERING CASE STUDIES / DEUTSCHE TELEKOM

Broadband Systems, Operational Confidence

My software engineering work in broadband provisioning and support applications



Gerry Walter

Senior Applied AI Engineer

Full Stack Engineer | May 2017 - October 2019

Software I worked on. Engineering knowledge I bring.
A personal portfolio of my contribution and experience.

Cover: AI-generated conceptual editorial illustration, not an application screenshot.

START HERE

Introduction and Contents

A compact case study for technical and stakeholder presentations.

At Deutsche Telekom, I worked as a Full Stack Engineer on business-critical applications in the broadband environment. My work included Java EE/Spring backend services, user-facing functionality, secure enterprise integrations, modernization and production stabilization. FlexProd and SKS-ES were part of my application experience.

I contributed to application development while working with engineering, operations and business colleagues. The role combined implementation with defect analysis, testing and maintainability improvements. It gave me practical experience in software where a user action is connected to an operational process and where the meaning of a service result matters.

Contents

01 **What I Worked on at Telekom** 3

02 **FlexProd: My Provisioning Context** 4

03 **SKS-ES: My Diagnostic Application Context** 5

04 **My Integration and Modernization Practice** 6

05 **My Work in a Large-Scale Environment** 7

06 **How I Worked on Reliable Delivery** 8

07 **Glossary and References** 9

08 **Closing Perspective** 10

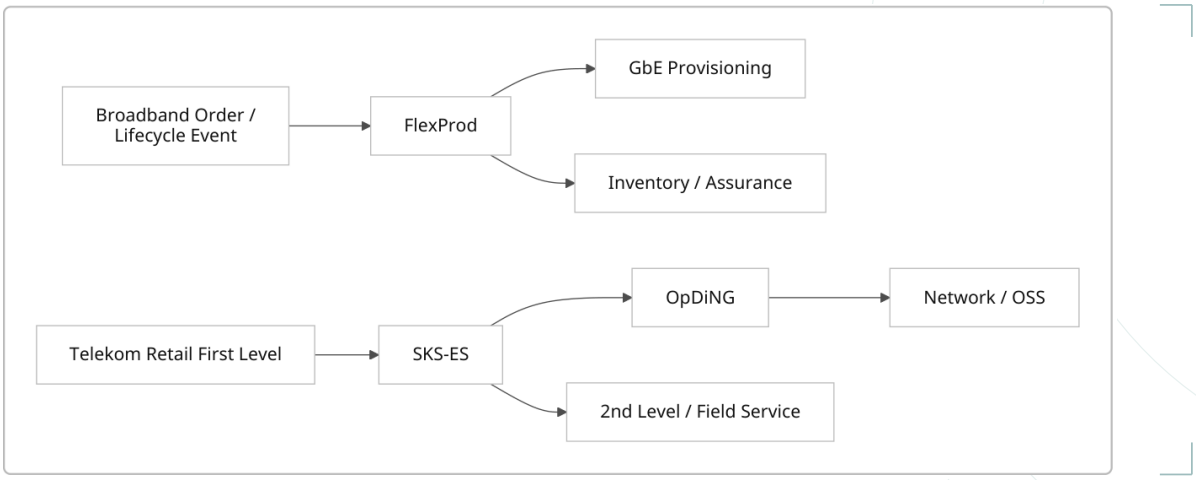
My engineering approach

I bring an operational perspective to software engineering. I want to understand the requested action, the systems involved and the outcome the user needs to see. My Telekom experience developed that perspective through hands-on application work, enterprise integration and collaboration with the teams responsible for ongoing operation.

CHAPTER 01

What I Worked on at Telekom

Full-stack development in a business-critical broadband application environment.



01 The application context of my work: FlexProd for provisioning and lifecycle processes; SKS-ES for retail support diagnostics. Simplified landscape view.

Hands-on application development

I developed backend services and user-facing functionality using Java EE, Spring and web technologies. My work supported business-critical applications and included secure enterprise integrations. I worked across the application rather than focusing only on the interface or only on backend implementation.

FlexProd and SKS-ES

My application experience included FlexProd and SKS-ES. These systems provide two useful perspectives on broadband operations: supporting the provisioning and lifecycle of a service, and helping support teams investigate its condition. Understanding that operational purpose gave the development work a concrete business context.

Modernization and maintainability

I contributed modernization work to improve application structure, maintainability and reliable workflows. This required working within existing software and integrations while making it easier to develop and support. I bring experience evolving enterprise applications, where a technical change has to fit the surrounding system and ongoing operation.

Working with operational teams

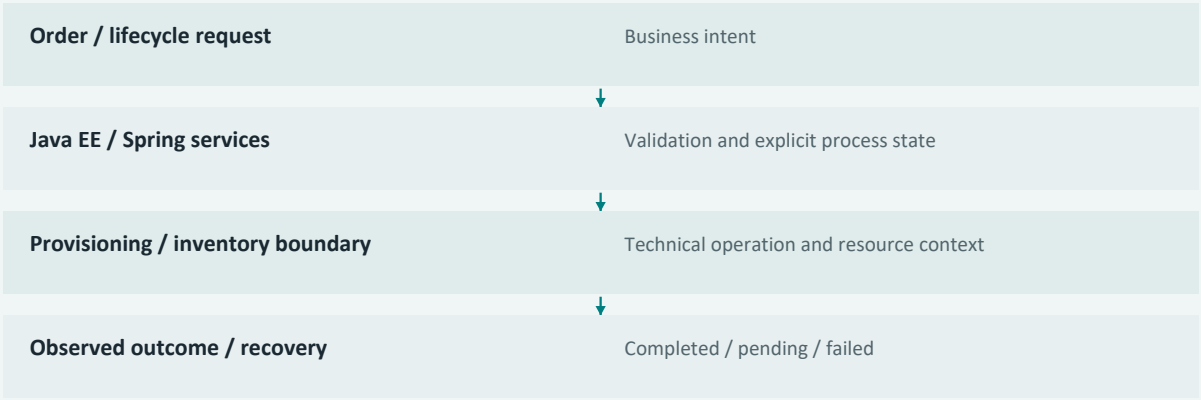
I contributed defect analysis, testing and production stabilization with engineering, operations and business colleagues. This connected implementation to the people responsible for using and running the software. It also developed my ability to relate a visible problem to the application behavior and integrations behind it.

Java EE / Spring / Web development / Enterprise integration | References: p. 9

CHAPTER 02

FlexProd: My Provisioning Context

Understanding the process behind the service implementation.



02 Provisioning connects a requested service change with technical work and an operational result. The diagram explains responsibilities rather than internal deployment details.

Understanding what the system does

FlexProd provided the provisioning and lifecycle context for part of my Telekom work. Its historical role covered broadband Double/Triple Play provisioning on a Gigabit Ethernet platform, with inventory and assurance concerns. That domain knowledge helped me understand the operational process behind application functionality rather than viewing it as a collection of unrelated service calls.

My application engineering work

My contribution involved Java EE/Spring backend development, user-facing functionality and enterprise integrations. I worked on business-critical application behavior within an established environment. This required connecting implementation details to the workflow the application supported and considering how changes would affect the surrounding software.

Thinking in lifecycle states

Provisioning sharpened my understanding of the difference between a requested action and a completed operational outcome. An application needs to communicate the state of the process, not merely whether it sent a request. This is part of the systems knowledge I bring to integration work and to the design of user-facing behavior.

Maintaining software in context

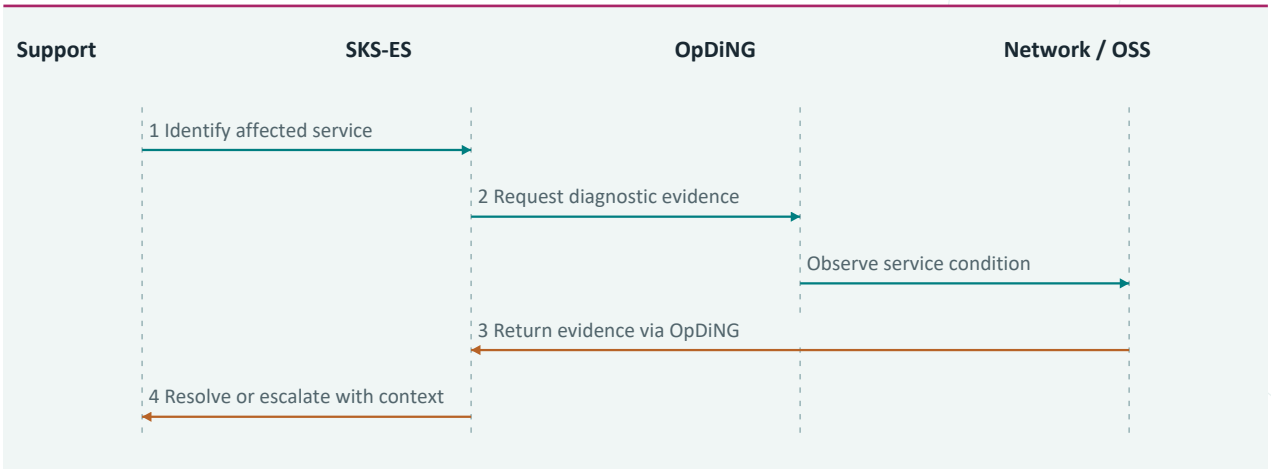
My modernization work focused on structure, maintainability and reliable workflows. In an operational application, an improvement has to fit existing processes and integrations. I learned to value changes that make behavior easier to understand and support, alongside the implementation of new functionality.

Provisioning / Application lifecycle / Java services / Maintainability | References: p. 9

CHAPTER 03

SKS-ES: My Diagnostic Application Context

Connecting support-facing functionality with technical service information.



03 SKS-ES connects retail support to OpDiNG diagnostics. Simplified sequence to explain the application domain, not a claim of ownership of every network service.

Understanding the support task

SKS-ES was part of my Telekom application experience in the first-level support environment. Its diagnostic role connects the support interaction with technical information about the affected service. That context matters to me as a developer: the result has to help someone understand a problem and decide what to do next.

The integration behind the interface

SKS-ES connects to OpDiNG in the diagnostic landscape. This is a concrete example of the relationship between a user-facing application and the operational systems behind it. My own work included backend services, frontend functionality and enterprise integrations, giving me experience following application behavior across those kinds of boundaries.

Distinguishing a result from resolution

Diagnostic information is useful because it supports a decision; receiving a response does not necessarily mean a service problem is resolved. I bring that distinction to the way I think about application state and user feedback. The interface and the integration need to be understood together, in the context of the support workflow.

Working on dependable behavior

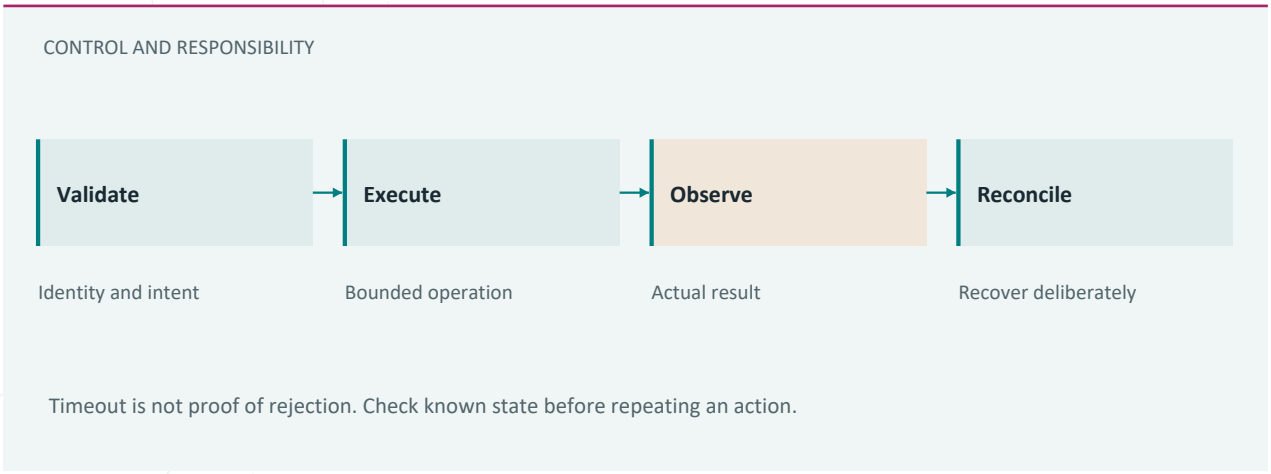
My Telekom role included defect analysis, testing and production stabilization. In a support application, these activities are closely tied to operational usefulness. I worked with engineering, operations and business colleagues, connecting technical investigation with the behavior that users needed from the application.

SKS-ES / OpDiNG context / Full-stack development / Diagnostics | References: p. 9

CHAPTER 04

My Integration and Modernization Practice

Working across established services without losing the operational purpose.



04 The systems perspective I bring: understand the requested operation, follow its execution and make its result meaningful to the application.

Enterprise integration experience

I developed secure integrations alongside Java EE/Spring services and user-facing functionality. This work required understanding connected application behavior rather than only the local method being implemented. I bring practical experience with the responsibility that sits between a user action, a service request and an operational result.

Reasoning about incomplete outcomes

Operational software has to distinguish what was requested from what actually happened. A delayed or missing response is not automatically a completed failure or a successful action. This is an important part of my integration knowledge: recovery decisions depend on understanding system state, not simply repeating the last request.

Improving maintainability

I contributed to modernization aimed at improving structure, maintainability and reliable workflows. Working in an existing enterprise environment requires attention to dependencies and behavior that other teams rely on. I value clear boundaries and understandable implementation because they make both future development and production investigation more effective.

Connecting development and operation

Production stabilization was part of my role, alongside development and testing. I worked with colleagues from engineering, operations and the business. That experience shaped my approach to software: a feature is part of an ongoing operational process, and the engineer needs to understand more than the code path that initially appears to work.

Secure integration / Modernization / Operational state / Stabilization | References: p. 9

CHAPTER 05

My Work in a Large-Scale Environment

Personal engineering contribution within Telekom's broadband business.

13.730m
retail broadband lines

17.479m
retail IP-based lines

8.529m
retail FTTx lines

My implementation	**Java EE / Spring**	Backend and user-facing functionality
My application context	**FlexProd / SKS-ES**	Provisioning and diagnostics
My delivery work	**Testing / stabilization**	Modernization and integration

05 Telekom Germany at year-end 2019: corporate operating context, not application throughput or a personal performance result. Sources and category definitions are on page 9.

The scale around the work

At year-end 2019, Telekom Germany reported 13.730 million retail broadband lines, 17.479 million retail IP-based lines and 8.529 million retail fiber-based lines. These figures describe the business environment in which I worked. They underline why continuity and operational reliability mattered, without assigning the entire national footprint to an individual application.

My contribution within that setting

My responsibility was hands-on software engineering: backend services, user-facing functionality, secure integration, modernization, testing and stabilization. I contributed to those areas with engineering, operations and business colleagues. The portfolio value is the practical work I did in that environment, rather than a claim to have owned the whole network.

Two complementary application domains

FlexProd and SKS-ES connected my experience to provisioning and diagnostics. Working in these domains developed my understanding of intended service state, observed conditions and the information needed for an operational decision. That systems perspective complements my Java EE/Spring and full-stack implementation experience.

The engineering value I bring

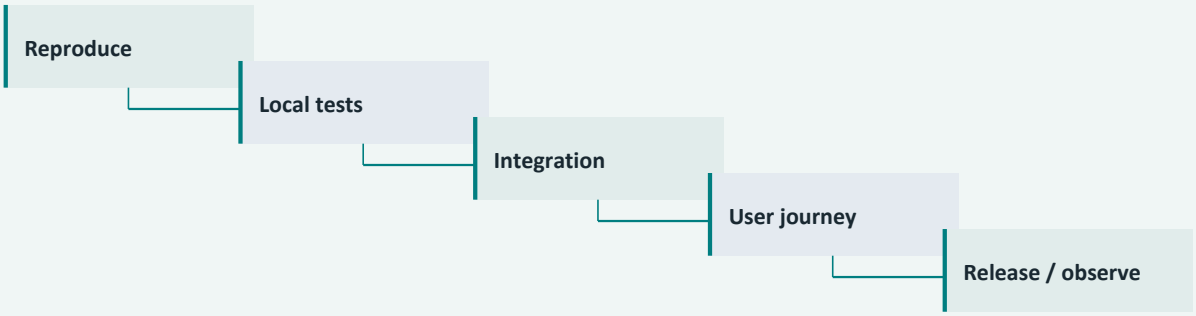
I can work in established business-critical software, understand its operational purpose and contribute changes across the application and integration layers. I also bring experience with defect investigation and production stabilization. Together, these capabilities let me connect technical implementation to the reliability and maintainability a team needs over time.

Personal contribution; Telekom Germany 2019 context: references p. 9 | References: p. 9

CHAPTER 06

How I Worked on Reliable Delivery

Implementation, testing and stabilization as connected responsibilities.



Failure examples belong in regression tests. A release needs an observable outcome.

06 My work combined development, defect analysis, testing and production stabilization with engineering, operations and business teams.

Implementation with maintainability

I worked on backend services, user-facing functionality and modernization in Java EE/Spring applications. Maintainability was part of the responsibility, not an optional step after implementation. I contributed changes to existing software while considering how the application would continue to be developed and supported.

Testing and defect analysis

I contributed testing and investigated defects in business-critical application behavior. This work connected the user-visible problem with the implementation and integration context. It strengthened my ability to reason about a workflow across layers and to relate a technical correction to the behavior the application needed to deliver.

Stabilization with the wider team

I worked with engineering, operations and business colleagues on production stabilization. That collaboration gave me a broader view of quality than the local development environment alone. Understanding how a system is used and supported helps me make better implementation decisions and contribute more effectively when a problem crosses team boundaries.

A foundation for my applied AI work

My Telekom experience is part of the software engineering foundation I bring to applied AI. I understand the importance of explicit application state, dependable integrations and operationally meaningful outcomes. I use that perspective when connecting intelligent capabilities to real workflows, while retaining the delivery discipline required of enterprise software.

Application development / Testing / Defect analysis / Production stabilization | References: p. 9

REFERENCE

Glossary and References

Technical terms, keyword page references and supporting project context.

Supporting References

[CV] Gerry Walter: career and project history

Personal role, dates, technology stack and contribution details supplied by the author.

[R] Project research and contextual references

Supporting research supplied by the author, 12 September 2026. Distinguishes application history and external business context.

[T1] Deutsche Telekom: Annual Report 2019

Germany customer-development table, 31 December 2019. Business scale, not application load. Verified September 2026.

report.telekom.com

[T2] TUV Rheinland / Bundesnetzagentur

Equivalence of Input report, 8 July 2020. SKS-ES/OpDiNG diagnostics and the examined FTTH-FB process; after the engagement.

www.bundesnetzagentur.de

[F1] Historical FlexProd role description

Mirrored Telekom job description cited in supplied research: GbE broadband provisioning and lifecycle context. Secondary evidence; not a current stack inventory.

en.wizbii.com

Glossary / Keyword Index

FTTx / pp. 7

Fiber access family; includes FTTC/VDSL and FTTH, not only fiber to the home.

FlexProd / pp. 3-4

Historical broadband provisioning/lifecycle application context.

ONT / pp. 5

Optical network terminal at the customer-side optical connection.

OSS / pp. 4-6

Operations support systems for network/service processes.

OpDiNG / pp. 5-6

Telekom diagnostics system reached by SKS-ES.

SKS-ES / pp. 3, 5

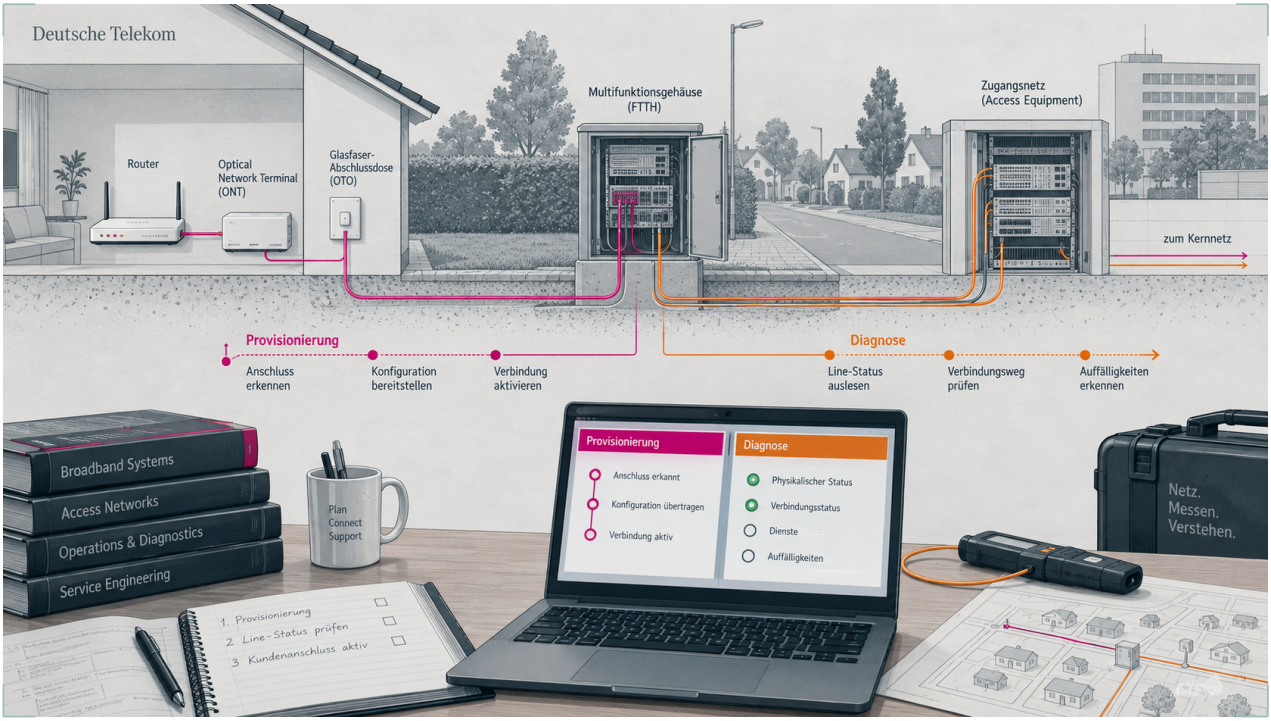
Service Kompetenz Steuerung-Expertensystem; retail first-level diagnostics.

Visual provenance

Cover and closing artwork: newly AI-generated conceptual illustrations. Technical diagrams: original, editable explanatory drawings. Research excerpts: cropped from the user-supplied PDF where indicated. None is presented as a private application screenshot.

CLOSING PERSPECTIVE

I build with the operational outcome in mind.



At Deutsche Telekom, I contributed full-stack development, Java EE/Spring services, enterprise integration, modernization and production stabilization. FlexProd and SKS-ES gave that work a concrete operational setting: broadband provisioning and the applications supporting service diagnostics.

I bring this experience into my work as a Senior Applied AI Engineer. It has shaped how I think about system state, integration behavior, user-facing outcomes and recovery. Those are practical software engineering responsibilities, whether the next capability is implemented as a conventional service or an AI-enabled workflow.

Gerry Walter

Senior Applied AI Engineer

Deutsche Telekom / Personal engineering portfolio / September 2026