

Bridging in a Nutshell

We bring business, IT and operations together to innovate your end-to-end processes

Since more than 15 years, we here at Scheer E2E have developed world-class methods and tooling to help organizations with medium to large scale IT infrastructure gear up for rapid - and easier - change. Our secret is simple: We empower business and IT specialists to collaborate more effectively on business-driven process integration efforts.

In this introduction, we discuss the fundamental mechanisms of what we call bridging. These are the mechanisms which will make it easier for you to bring innovation and successful change to your organization. Regardless if you're an IT manager, a business analyst or a developer, "Bridging in a Nutshell" provides you with just enough understanding - and motivation - before you want to dive into more than 2'000 pages of freely available online documentation and self-study training material here at doc.scheer-pas.com.

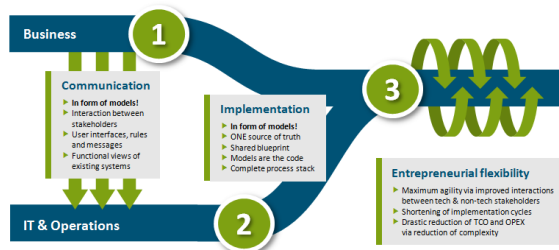


Figure 1: Scheer E2E facilitates a joint approach between business, development and operations to make change simply easier.

Obstacles for business innovation

Conducting business in a networked economy forces companies to reinvent their core processes. Customer purchasing behavior has changed in favor of flexible choice and 7x24 shopping sprees on the Web at home or on a mobile. Supply chains have become increasingly globalized. Business commodities like Cash Collection, Return Logistics and a myriad of other capabilities are no longer considered core and more and more farmed out to external Cloud providers.

Not long ago, few companies saw more in the Internet than a marketing tool to reinforce their brand. The combination of acceptance, technical progress and economic pressure has changed this view. Terms like B2B, B2C, social CRM and multi-channeling have taken center stage in building and maintaining a close dialog with customers, while straight-through-processing and end-to-end process optimization aim at improving and controlling operating margins.

The top 5 challenges are always the same

Responding to external changes requires a high degree of technical *and* organizational flexibility – which is usually not a given. Many companies are trapped in their legacy application silos, operating mostly in batch mode. Basic internal procedures are still carried out manually, in particular at divisional and company boundaries.

Let's not forget the financial aspects. As if the economic crisis had not taken its toll already, desperately needed funds for innovation compete with the high running cost of daily business. And finally, change means risk and the looming danger of interrupting ongoing operations.

Not understanding the status quo is the most serious obstacle to successful change

Closer examination of these challenges readily points to the same 5 obstacles standing in the way of effective business transformation, and they sure sound unpleasantly familiar:

- **Misunderstandings are the norm** – Business and IT lack a common language and mutual trust. To be successful, all stakeholders must be on board, or else, the transformation will be nothing more than just another infrastructure project.
- **Critical information is inaccessible** – It's locked away in the brains of key people, instead of being readily available in form of accurate, up-to-date documentation. Time-to-market hinges on how available these individuals are – and usually they are not.



Got time for a new approach?

On this Page:

- We bring business, IT and operations together to innovate your end-to-end processes
- Obstacles for business innovation
 - The top 5 challenges are always the same
 - Calling for a radically new approach
 - Leveraging the Cloud opportunity: the Nikon case
- Gearing up for rapid - and easier - change
 - Bridging Principle 1: Involving all stakeholders
 - Bridging Principle 2: A single source of truth
 - Bridging Principle 3: Top-down transparency
 - Bridging Principle 4: Clear separation of concerns
 - Bridging Principle 5: Simplification by abstraction
- Tooling for success: the BRIDGE platform
 - Success Factor 1: ONE tool only
 - Success Factor 2: ONE context, TWO views
 - Success Factor 3: HIGH performance, SMALL footprint
 - Success Factor 4: Debugging for distributed environments
 - Success Factor 5: ALL-in-ONE SOA + implicit governance

- **Total cost is out of balance** – 80% of the total bill for information systems is going to maintenance and operations, a fact which most have come to accept as a given. But in reality, this cost structure is the consequence of dated development and integration methodologies.
- **Governance is not for free** – The transparency required by modern business practices, especially the compliance with legal regulations, causes additional overhead, adds more cost and dampens productivity. But – shouldn't transparency lead to higher efficiency?
- **Infrastructure reuse is a must** – There is not enough time and money to start from scratch. Some of the capabilities can be replaced by external Cloud offerings, but clever reuse of existing applications is a necessity, not an option. Reuse will also maximize the ROI on previous investments.

Calling for a radically new approach

During the past 15 years, we at Scheer E2E have faced the same obstacles and learned to deal with them successfully. Practical experience led us to establish an original, unique and exceptionally effective approach to eliminate all of them in one go. Does this sound too good to be true? In any case, based on our unique approach, we can look back to more than 100 successful business transformation efforts. Explaining how this unique approach works is the topic of this introductory overview.

In the following paragraphs, we invite you to join us on a journey to higher levels of organizational productivity. We will introduce effective techniques to hold the "reality factor" at bay which – despite careful precautions – tends to creep into all real life business initiatives. To that extent, we will discuss **five principles** which will help you stay clear of the above mentioned obstacles, along with some comments how these principles can be delivered. We will present an **innovative tool set** – the **Scheer PAS BRIDGE** platform – which implements these principles. A **real-life case anatomy** will demonstrate how you can get started to apply the approach in your own company. Finally, we will discuss the **results you can expect, measured terms of faster time-to-market, reduced total cost and gained business flexibility**.

But before we dive into the details, let's provide a sneak preview of what's ahead, by looking at a concrete case which demonstrates how Scheer E2E's approach supports real-life business transformation initiatives.

Leveraging the Cloud opportunity: the Nikon case

The case is about a scenario that is quite common these days: how to combine existing applications with online private and public Cloud solutions for business commodities, to gain flexibility and reduce operational cost. The company in question is Nikon, a well-known global brand specializing in optics and imaging, with USD 10bn in turnover. For its European subsidiaries, the company is presently overhauling its complete B2C and B2B strategy. In the past, all interactions with dealers were managed via its country-based operations, predominantly via Fax and Email. This resulted in high administrative overhead for order management, leaving dealers stranded with long cycle times to process order change requests. This model did not scale – a major obstacle for further growth.

Nikon chose a service-oriented approach to redo its supporting business information backbone, effectively combining reuse of legacy applications with modern Cloud-based offerings to accelerate the implementation of the business transformation effort. The main objective was to achieve a high degree of automation across the end-to-end order management process, and to facilitate better scalability for revenue generation across all sales channels. At the same time, the company wanted to increase its flexibility to react to country specific needs. It also wanted to be prepared for changes in the provider landscape, e.g. for Cash Collection or Return Logistics. As part of the transformation effort, country organizations were refocused on expanding the dealer network, instead of spending their time with administration and operations, a change more compatible with rapid future revenue growth.

To accelerate its transformation effort, the company applied the approach outlined in this white paper and used the BRIDGE platform as the core Business Integration engine to control all information exchange between the central SAP system and the Cloud-based Order Entry, Payment Processing and Return Logistics (see Figure 2).

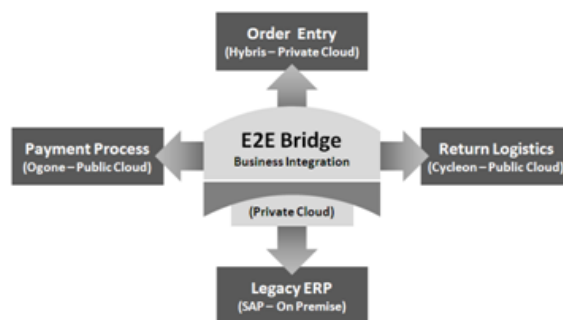


Figure 2: Bridging between on-premise, private and public Cloud systems at Nikon.

With the central SAP system firmly ingrained into the Nikon's internal processes, its replacement would have been too costly, too time-consuming – and too risky. To effectively reuse ERP core services, the BRIDGE was used to wrap the custom SAP installation, so it could communicate effectively with online Cloud services. With an expected traffic of several million transactions per day, triggered by up to 27,000 online dealers, the BRIDGE was configured for high availability. To avoid exposing the SAP system to high fluctuations in transaction volume that could cause unexpected service interruption, the BRIDGE was also used to buffer all transactions in an effort to limit the transaction load on the ERP system. Decoupling the ERP from all other e-commerce application components also reduced the IT risk for future SAP upgrades, which was impressively demonstrated when the SAP system had to be taken down for a 2-day upgrade - while the BRIDGE allowed the shop to continue operations during the SAP downtime.

Three factors governed the selection of the BRIDGE for the Nikon initiative:

- **Transparency** – As we will see, the BRIDGE uses a purely model-based approach to foster unambiguous communication among all stakeholders during specification, implementation and day-to-day business operations. **This effectively eliminates misunderstandings and it reduces time-to-market, requiring fewer iterations.** For the Nikon case, the BRIDGE ensured architecture and processes stayed aligned with business needs, by creating a shared visual record of the company's new multichannel e-Commerce blueprint. The transparency of models helped also with the security aspects of the Cloud integration, ensuring all policies for information access were properly enforced, even across different Cloud providers. In other words, even though the company's business information was externalized, Nikon kept full control over who accessed what information and when.
- **Productivity** - Time-to-market was absolutely critical for Nikon. A long-drawn product selection process had delayed the start of the project for more than 5 months. You will learn shortly, that **the BRIDGE has a unique capability to keep business specification, technical design and the actual production systems automatically in sync at all times.** This, combined with a high degree of automation, which accelerated development of hundreds of interfaces, permitted such an aggressive project schedule, that the time lost during product evaluation was almost compensated for. Project implementation was shortened by several months, allowing Nikon's first European subsidiaries to go live just 4 months after starting the initiative. Based on the synchronicity of documentation and production environment, the productivity increase to implement change requests was substantial.
- **Performance** – The BRIDGE has unusually modest system requirements. For this initiative, this meant that – compared to more traditional infrastructure software – **only a fraction of the server resources was needed, which drastically reduced the operating cost.** At the same time, the BRIDGE can easily scale to billions of transactions per day, which made it a future-proof investment for Nikon.

Gearing up for rapid - and easier - change

Having reviewed the main obstacles for IT-driven business innovation, we will now discuss five key principles that can be employed to eliminate these obstacles and raise productivity levels. In turn, these principles serve to introduce the BRIDGE platform, which was built to apply these principles to concrete business situations. **In this context, you will see that the term “bridging” refers to three different aspects: the bridging between people in terms of communication among stakeholders, the bridging between systems in terms of connectivity, and finally, the bridging between stages of evolution in terms of managing incremental change.**

Bridging Principle 1: Involving all stakeholders

Models play an important role in the description of complex scenarios and are used extensively in the development of information systems. They are well suited to serve as a **communication** vehicle between collaborating individuals of different professions – a visual “lingua franca” that offers a common basis of understanding. They convey information more effectively than code which can only be understood by developers. Effective use of models for interdisciplinary communication is therefore an essential ingredient to involve all stakeholders in the continued evolution of information systems (see Figure 3).



Figure 3: Models are better than code to communicate ideas.

Models eliminate misunderstandings by serving as a common language between business and IT.

It is established practice to use models during process analysis, for high-level design of information systems, as well as description of data structures and their interrelations. The BRIDGE expands on the traditional use of models by using them **throughout the entire lifecycle from specification to operational use** to describe:

- Processes, user interfaces, services, data, roles and access rights
- Workflows, business logic, service composition and orchestration
- Semantic transformation of metadata between applications
- System landscape specification and infrastructure configurations
- Testing, root cause analysis, profiling and transaction monitoring

Using models to describe all aspects of information systems from specification to operations, the BRIDGE effectively removes the tool gaps along the lifecycle which are known from more traditional development environments (see Figure 4).



Figure 4: Using models to bridge the tool gaps in the lifecycle.

Bridging Principle 2: A single source of truth

Using models throughout the lifecycle provides an opportunity to employ them for more than just specification and documentation. In fact, models can be directly transformed into running programs.

Traditional software development environments make more and more use of such model transformation to increase software quality. They do so by producing code in another high-level programming language, such as Java or C#. However, in most cases this form of code generation covers only certain aspects of development, and manual steps are needed to produce a running system.

While this represents certainly a productivity enhancement, such model transformation leads to three kinds of problems:

- **Lack of performance** – Automatic model transformation often leads to systems with very poor performance and scalability
- **Synchronization challenge** – Manual code additions need to be kept in sync with the original design models, and if neglected, inconsistencies occur which make further use of models difficult
- **Error detection** – Elimination of errors has to be performed at the generated code level, which is different from the design model which served as input. This is difficult to understand, because detecting a one to one relationship between model and code is nearly impossible.

These problems can be solved with the introduction of a more rigorous and complete approach that skips intermediate code generation all together, producing a running system from the models without any manual intervention. This concept is referred to as **Direct Model Execution**.

The BRIDGE is one of the first platforms worldwide to apply this concept. In doing so - and by exclusively using modeling standards such as UML and BPMN instead of proprietary modeling notation - it provides an implicit insurance policy for keeping documentation and production systems in sync without manual governance steps. In fact, **the documentation is the code**, and therefore, **the specification is the production system, serving as a single source of truth across the entire lifecycle** (see Figure 5).

Direct Model Execution eliminates the dependency on individuals by keeping specification and production systems permanently in sync and making them accessible to all stakeholders from business, development and operations.

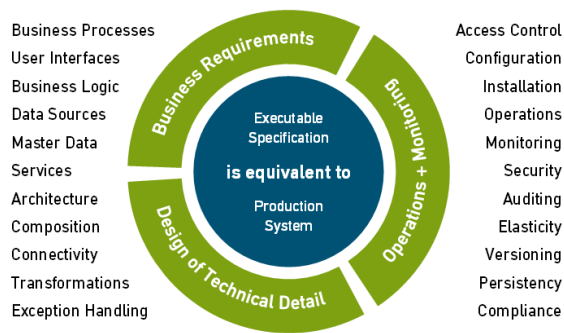


Figure 5: The specification IS the production system.

With all potential discrepancies between design view and implementation removed, **the BRIDGE offers a powerful platform for incremental system improvement, where all stakeholders can collaborate with the certainty of understanding what is currently used in production.**

Bridging Principle 3: Top-down transparency

Using models across the full software stack increases flexibility and fosters reuse.

Model Execution is gaining rapid acceptance at the process level, providing drastic improvements in enterprise level agility. But with the use of models constrained to the high-level views, too many gaps persist between the executable processes and the actual systems which serve as data sources for these processes, causing major governance hurdles.

To eliminate these gaps and achieve complete top-down transparency, the development and maintenance of processes, services and interfaces must be combined into one common platform. This is the approach which was chosen by the BRIDGE to provide better contextual information (see Figure 6).

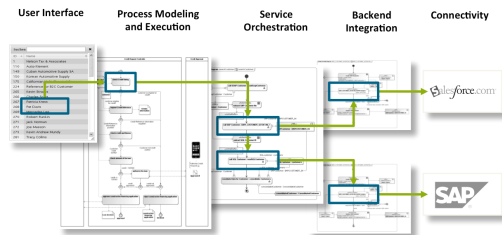


Figure 6: Process integration with SAP and a Cloud-based Salesforce.com service.

Bridging Principle 4: Clear separation of concerns

A model-based interceptor reduces governance cost by simplifying compliance with non-functional policies.

Apart from the actual business functionality, any information system covers non-functional aspects to guarantee its proper application. Such non-functional aspects include security, in terms of who can do what and when, technical aspects, such as scalability and performance management, but also business relevant aspects, e.g. the extraction of key performance indicators, micro-billing or the safe handling of transactions.

In traditional information systems, functional and non-functional elements are often mixed, making it difficult to change either during change management, causing frustration and unnecessary additional cost. In order to remedy this problem, the BRIDGE introduces a powerful concept that allows permanent separation of functional and non-functional aspects.

The BRIDGE offers this separation using a transparent **interceptor approach** (see Figure 7). As a result, policies can be changed at any time without impacting the functional side of the information system.



Figure 7: Separation of concerns using an interceptor approach.

In combination with direct model execution, this interceptor approach represents a powerful **framework for policy management**. Since compliance relevant aspects are executed as modeled, their separation from functional elements dramatically simplifies external audits. Furthermore, isolating all non-functional aspects into their own realm also reduces ramifications on functional development. **As can be expected, this leads to substantial reduction of time-to-market, while eliminating frictions and misunderstandings between functional and non-functional development teams.**

Bridging Principle 5: Simplification by abstraction

Abstraction is a powerful mechanism to exploit commonalities among a collection of individual concepts. By hiding complexity that may well be present at a lower level, abstraction leads directly to simplification. One popular application of abstraction in modern day IT is the **virtualization** of computing environments. It provides platform independence from the underlying operating environment, granting identical behavior of a business application on a variety of systems and less vendor lock-in.

In addition to platform virtualization, the BRIDGE pushes the state-of-the-art in using abstraction to cover an even broader range of concepts, including **connectivity, persistence and packaging**. Whereas the latter two offer more convenience for IT specialists, abstracting connectivity has powerful implications at the business level and deserves a closer look.

A common representation of all backend types reduces complexity, facilitates reuse and reduces cost of change.

Imagine you want to integrate SAP with Oracle. At the technical and the business level, their interfaces are substantially different. To simplify integration, the BRIDGE introduces a **generic backend abstraction**, allowing you to deal with both interfaces in exactly the same way. In addition to SAP and Oracle, the BRIDGE offers this capability for 60+ types of backend systems. This means it can act as a **universal adapter** which hides the details of individual backend technologies.

This unique capability is made possible by two mechanisms the BRIDGE employs to facilitate this simplification. The first is a **common structure** for representing backend systems within the executable model. The second mechanism is a fully **automated analysis** of existing backend interfaces.

In addition to simplifying the management of application interfaces, and therefore reducing the total cost of ownership on a per interface basis, this mechanism also facilitates the migration and consolidation of backend systems, providing an effective infection shield that can be used temporarily as well as permanently.

Tooling for success: the BRIDGE platform

Now that the basic principles of the all model execution methodology are established, we turn our attention to the tooling that supports it.

Success Factor 1: ONE tool only

Description of the development lifecycle and all architectural components in form of executable models permits for a radically simple design choice: **one single tool** – the BRIDGE – to go from specification to production, covering all aspects of integrating systems, processes and user interfaces.

A single tool approach reduces complexity and provides more reuse.

Just like the two faces of a coin, the BRIDGE has two sides. On the one side, it provides an **authoring environment** called **Simplify**. It offers capabilities required for creation and maintenance of models, including validation, testing and transformation into an executable form. On the other side, we have the **BRIDGE**, which “executes” models in production (see Figure 8).

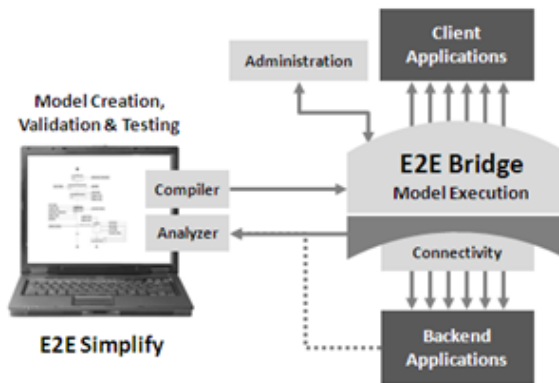


Figure 8: The BRIDGE platform ecosystem.

This single-tool-approach is quite different from traditional enterprise software infrastructure. With the approach traditional vendors have taken, a collection of several tools needs to be mastered to cover the same functional breadth and depth. This may include application servers, process engines, service buses adapter frameworks, user interface technologies, master data management tools and security policy management frameworks. The BRIDGE covers all these functions in one tool. With less moving parts, this remarkable simplicity – and flexibility – of the BRIDGE leads to a major boost of organizational productivity.

Success Factor 2: ONE context, TWO views

For business specifications, predominantly at the process level, the standard Business Process Modeling Notation (BPMN®) is used to present non-technical audiences with sufficient abstraction as they collect their requirements. High-level models can have informal character and serve as complementary documentation. At lower levels, where technical processes require the sourcing of information from individual services or applications, BPMN models represent an unambiguous design contract between business specification and technical design. At this level, BPMN models take on a more formal character, which is relevant for process execution.

Catering for different preferences: BPMN for the business, UML for IT specialists.

For the technical design, a subset of the Unified Modeling Language (UML®) provides technical users such as developers, security staff, test teams and operators with more expressivity, as required for rapid implementation of backend connectivity, application logic, process automation, and user interfaces. Diagramming capabilities address the complete functional design, including structural models to represent data and mappings, dynamic models for process and service behavior, and architectural models to define packaging and deployment schemes for production use.

The authoring environment houses also a full suite of importers that allow existing application interfaces to be analyzed, wrapped and re-documented in the UML standard. Out-of-the box, some 60+ interface types are presently supported, which caters for most known integration scenarios.

Essential for the functioning of the all model-driven approach is the translation of the specification models into something that can be executed, just as you would with a regular software program. This task is performed by a Compiler (see Figure 7 above), which is an integral part of the authoring environment. In addition to transforming models into their executable form, the Compiler is also in charge of model validation, consistency checking, automatic performance optimization, as well as packaging for and deployment into the production environment.

Success Factor 3: HIGH performance, SMALL footprint

The BRIDGE owes many of its key characteristics to the challenging requirements of the financial services industry, where it was first deployed, most notably when it comes to transaction volume, resiliency and security. As a result, the BRIDGE constitutes a very flexible, high-performance execution infrastructure that can cope with Xtreme Transaction Processing. Furthermore, it can be configured for a diverse range of operational needs.

Infrastructure efficiency reduces total cost of ownership.

Capable of handling billions of transactions per day, the implementation and design of the BRIDGE is highly resource efficient. In addition, it has a very small system footprint. This reduces the server cost per transaction – often by more than an order of magnitude. For average size deployments, e.g. for a mid-size organization with less than a million transactions per day, this means that the BRIDGE can be co-

installed on a server that is already in used by another application, e.g. an e-Commerce system. This is a major factor in reducing total cost. Following the same logic, the BRIDGE can be effectively operated in a Cloud environment, or even as an on-premise appliance.

One of the reasons for this high resource efficiency is the fact that the BRIDGE runs directly on the – real or virtual – operating system, requiring nothing else but the operating system to function properly. So far, it has been ported to more than half a dozen different operating systems, providing a common abstraction layer for executable models, resulting in 100% platform independence. I.e. if you decide to move from an initial BRIDGE installation on Microsoft Windows® to a more scalable Linux® environment, you can do so without any functional modifications, a great operational advantage.

There might be a variety of reasons for operating multiple BRIDGE instances, e.g. if compute resources are distributed across multiple countries to provide local operational independence, or for dynamic scaling and load-balancing as part of an elastic Cloud infrastructure, or to provide redundancy and failsafe behavior that limits operational risk. Any such configurations are conveniently described as UML models. Subsequent deployment follows these models and is fully automated.

Success Factor 4: Debugging for distributed environments

If you ever had to track down the reasons for a failed transaction in a heterogeneous application environment, chances are you know what it means to suffer. Traditional software development tools provide formidable mechanisms to chase down erroneous behavior within a single application, but they leave you completely stranded when it comes down to chasing problems that occur between applications.

Effective root cause analysis in distributed environments drastically reduces time-to-market.

To address this common challenge – the nightmare of virtually every integration project – the authoring environment of the BRIDGE offers a powerful tool, the **Scheer PAS ANALYZER** (see Figure 8), which is specifically conceived to make quality assurance in distributed, heterogeneous environments as transparent as possible. It fulfills three main functions:

- **Automatic test case generation** – For every process, service or interface, the ANALYZER automatically creates a test case, which can then be fed with variety of test inputs to ensure proper functioning of the component in question. Such test input can be automatically gathered by recording business data from production use. Latter capability can be used to increase test coverage before major upgrades or modifications. All test cases can be conveniently grouped to form comprehensive scenarios for regression testing.
- **Model-driven root cause analysis** – Considered by many to be the killer feature of the BRIDGE ecosystem, the ANALYZER provides fully model-driven – static and dynamic – error detection capabilities (see Figure 9). This provides highly valuable insight into the behavior of any component developed with the BRIDGE, as it visualizes the complete data flow, it collects and reports status information from all surrounding systems directly within the model context, and it provides detailed profiling information to point out performance bottlenecks in the distributed application environment.
- **Monitoring and analysis of production use** - In concert with the BRIDGE, the ANALYZER can investigate individual transactions from production runs in case of operational issues. Within seconds, it permits exact localization of the problem and allows for corrective action. It can also provide detailed statistical analysis regarding production use of individual components, including their mutual dependencies. This information provides crucial insight to focus modernization efforts on those components that are actually used.

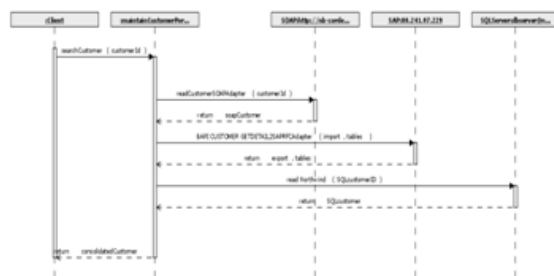


Figure 9: Model-driven root cause analysis.

Success Factor 5: ALL-in-ONE SOA + implicit governance

There are two scenarios how the BRIDGE can be used in practice: as a tool for communication, quality management and policy enforcement, and as an all-in-one platform for service-oriented development and integration.

An executable design contract allows for efficient SLA enforcement.

In the first scenario, the BRIDGE serves as an **Executable Design Contract** that can be used for collaborative specification and requirements engineering prior to, for quality assurance during and for acceptance testing after implementation, to enforce service level agreements in an unambiguous and transparent fashion. In this scenario, the role of the BRIDGE is to build up precise specifications for processes, services and interfaces, and to produce a test framework against which implementations based on other development tools can be validated.

This scenario is particularly useful in situations where an organization has outsourced its development to one or more external service providers. From an end user perspective, the Executable Design Contract is an effective way to limit project cost. For service providers, it provides a chance to better control their margin during fixed price engagements. Used this way, the BRIDGE provides a highly useful insurance policy – a real pain killer to keep complex projects on time and on budget.

The second scenario, the **All-in-One SOA Platform**, uses the full capabilities of the BRIDGE from specification to production use – a natural expansion of the Executable Design Contract introduced in the previous paragraph. Use of the BRIDGE as an All-in-One SOA Platform comes in three flavors, depending on the project characteristics:

- **System Integration** – The BRIDGE serves as last mile of connectivity for Application Integration, replacing manual interface coding to reduce total cost. In this use case, the BRIDGE is often used as adapter framework for middleware from other vendors, leading to higher ROI on previous investments.
- **Process Integration** – On top of System Integration, the BRIDGE can provide aggregation and orchestration of low level services into higher level business logic. Initiatives for B2B Integration, Process Control or multichannel e-Commerce fall into this category.
- **Business Integration** – Together with Process and System Integration, the BRIDGE can be used for the complete modernization of end-to-end processes, covering all layers from process to backend, including the development of user interfaces.