sandhata™

Technical Brief

Successful Service Virtualization

An introduction to how Service Virtualization can help IT to remain agile and deliver software faster at lower risk and cost

IT is constantly evolving around the increased complexity of business requirements. When the business needs to adopt lean, agile and componentised IT, many look to approaches such as SOA and similar distributed application architectures. Although these provide an effective model that adapts quickly, classical testing methodologies and tools do not fit these approaches well. This is where many businesses turn to service virtualization to achieve a faster time to market. As Service Virtualization is a relatively new solution, there are limited resources available. Although this often causes problems during the implementation phase, most of the problems can be resolved using a well-defined virtualization process.

Contents

What is service virtualization?

Service Virtualization simulates the behaviour of software components, to remove dependency constraints on development and testing teams, so they can deliver software faster, at lower risk and cost. In complex environments system dependency constraints can limit development, testing efforts and can significantly limit scope of testing – as well as compromise quality.

The purpose of this document is to give a brief overview of the Service Virtualization process, methodology and general best practices to successfully implement Service Virtualization.

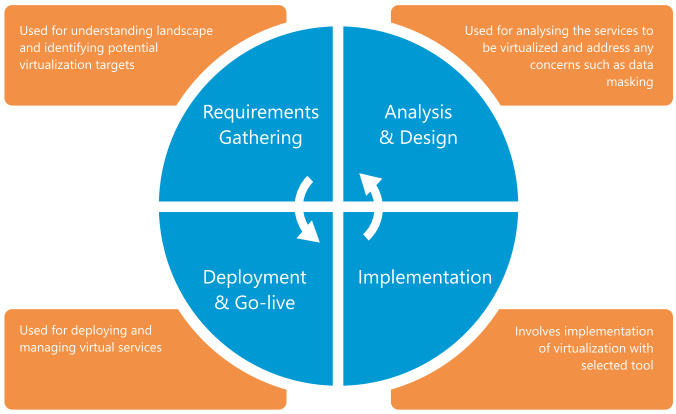
Methodology

Service Virtualization as a solution has huge potential. Unfortunately, IT will face numerous hurdles in its implementation. Most of these hurdles can be resolved by using a well-defined Service Virtualization Life Cycle, to help measure the progress of the implementation as well review it at logical points, to avoid major setbacks. This life cycle is well aligned with the agile life cycle of today's composite application approaches, as it is iterative and incremental. The Service Virtualization Life Cycle starts with a requirements management phase followed by analysis, design, and implementation – ultimately ending with deployment and management.

Requirement management phase

The main objective of this phase in the lifecycle is to identify what needs to be virtualized and to understand the proposed functional and non-functional requirements. Virtualization teams use this phase to understand the target environment landscape for virtual services. Once infrastructure requirements have been understood, the virtualization team then identifies the environment build strategies. There are always a few components in any environment that have very simple behaviours with low/no test data management complexities and negligible access constraints. These have low ROI and are therefore not ideal targets for service virtualization, except as a static service with single path for technology demonstrations. The ideal targets for virtualization are systems that have:

1. Dependency on an incomplete development
2. Access fees for testing
3. Inconsistent or invalid data
4. Physical external links
5. An inability to be tested independently



If the identified service is completely new and has no pre-existing data that can be found in production, then teams will need to determine how data would be sourced and ensure that artificial data can be created effectively. Virtualization teams needs to examine and consider all the components that consume the virtual service. If the service that needs to be virtualized already exists, then the data can be recorded.

There are two major goals you should be driving for when defining the requirements:

1. Determine the business purpose of the service
2. Understand the cost and benefit outcomes for the business and its intended users

Business purpose of the service

The fundamental premise of all virtual service planning is that a virtual service exists to serve a business purpose. There should always be a virtual service with "just enough" resource to continue testing instead of building a whole service.

Cost benefit

You might need to conduct more analysis of some - or all - of the following:

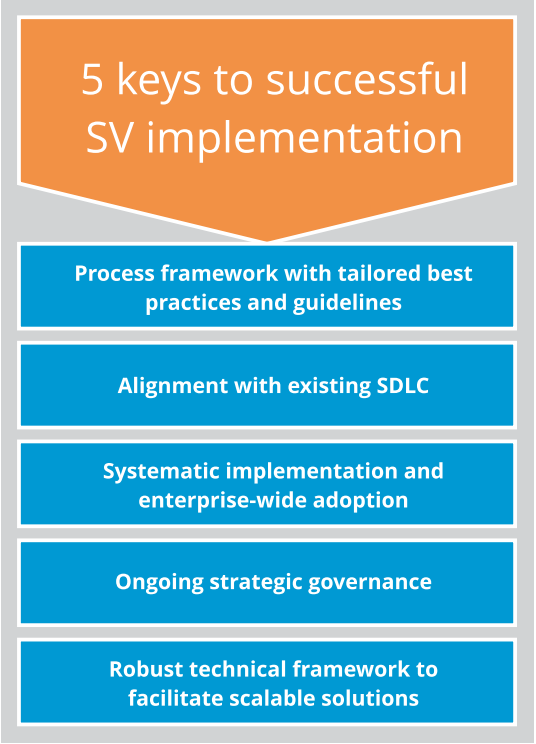
- Existing services, if any, and what they do
- Whether or not a virtual service is definitely required
- Whether or not existing services in the service-oriented architecture (SOA) can be leveraged, either partially or wholly.

Analysis and design phase

The virtualization team will analyse the requirements gathered in the previous phase, and then plan and document the process for building the virtual services. In this phase, teams focus on the services and components identified and their associated use cases. Teams use a variety of artefacts, such as functional specifications and sequence diagrams, to ensure complete coverage. By the end of the design phase, the virtualization team should ensure that they have all the valid input and output pairs for all scenarios in a given component. This can be achieved by asking the development team about request and response documents.

During the design phase it is important to:

1. Understand and clearly articulate the detailed requirements for the service. Make sure there is agreement between key players before the development starts.
2. Separate functional from non-functional requirements, and develop only to the functional requirements.
3. Try to utilise any similar services on the same transport if possible, rather than building a new service.



Implementation phase

The implementation phase involves the creation of virtual services, populating them with data and ensuring the quality of service by performing unit testing.

With the detailed requirements for the virtual service in place, the virtualization team can now decide to build the service from scratch, or extend a static service.

A static service has the technical framework established beforehand and can be extended to cater for the functional requirements.

The virtualization team initially builds this static service without any data. These services can then be used to derive test data by recording and masking the production information, or through manual data creation. Steps for recording data vary across products but generally they follow these steps:

1. Configure the components for recording
2. Modify the captured data to create an appropriate subset to cover scenarios
3. Mask sensitive data
4. Ensure all scenarios are covered

The virtualization team builds unit test cases to cover all the use cases and completes the unit testing. The team completes the implementation phase after ensuring that a virtual service can indeed perform in all of the identified use cases.

Deployment & management phase

Virtual services created in the previous phase are now configured and deployed in virtual service containers by the virtualization team. Most virtual services require minor corrections as they are used in the target environment. A well-designed virtual service allows for minor changes by incorporating the changes to the configuration or data – without changing the service from scratch.

About Sandhata

At Sandhata Technologies Ltd, we transform our customers' IT strategy by leveraging existing infrastructures to connect data, systems and processes to modernise and digitise today's business outcomes. We have proven experience as a partner to many of the world's most forward-thinking organisations within the financial, telecommunications, manufacturing, healthcare and retail industries. These organisations have got considerable investment in IT transformation programmes and a real desire to transform their IT to be future-proofed, digital-first and tailored to their business and customers' needs.

Our integration capabilities offer simple and flexible solutions to complex issues such as regulatory compliance requirements. The Sandhata integration platform can uplift middleware and provide information-driven services from a variety of data sources and systems. Our technical and process consultants provide great value addition by modernising the delivery of services to every customer it serves.

One step ahead

With a proven track record in implementing complex SV solutions as well as close relationships with market leading tool vendors, we are able to stay ahead of the curve in Service Virtualization.

1. Sandhata is a premium IBM business partner and the only service provider offering SV solutions across the globe.
2. We can deliver onshore/offshore and deliver managed service both on premise and in the cloud, based on requirement.
3. We understand the dependencies, complexities and challenges in Service Virtualization and can operate Centres of Excellence in centralised, federated and hybrid models.
4. Our service offerings provide our clients with both the tactical and strategic options to implement Service Virtualization.
5. Our processes and methodologies not only offer the best practices but also provide a framework for a successful and scalable implementation.

Experience is key

Our Experience in Service Virtualization goes back to when it was still an emerging technology, and we continuously provide a strong set of proven benefits to our clients.

- Improving quality
- Lowering on-going testing costs
- Enabling Testing earlier at component or interface layer
- Reducing reliance on application owners during integration testing
- Improving collaboration and efficiency between developers and testers to enable Agile Delivery - an evolution from Agile Development.
- Virtualizing end systems to remove dependencies, such as stub web services, mainframes, application servers and more.
- Observing events to accelerate defect investigation in the end-to-end system for grey box testing.
- Enabling quicker and frequent regression testing.
- Tailoring virtual services to be more efficient across the SDLC.

We Transform the Business of IT

For more information and to learn more about our services please contact us on:
+44 20 3300 1633

sandhata.com