

Microsoft® Operations Framework

Planning for Software-plus-Services:   
A MOF Companion Guide

Published: April 2009

For the latest information, please see [microsoft.com/mof](http://www.microsoft.com/technet/solutionaccelerators/default.mspx)

Copyright © 2009 Microsoft Corporation. This documentation is licensed to you under the Creative Commons Attribution License.  To view a copy of this license, visit [http://creativecommons.org/licenses/by/3.0/us/](http://creativecommons.org/licenses/by/3.0/us/%20) or send a letter to Creative Commons, 543 Howard Street, 5th Floor, San Francisco, California, 94105, USA.  When using this documentation, provide the following attribution: The Microsoft Operations Framework 4.0 is provided with permission from Microsoft Corporation.

This documentation is provided to you for informational purposes only, and is provided to you entirely "AS IS". Your use of the documentation cannot be understood as substituting for customized service and information that might be developed by Microsoft Corporation for a particular user based upon that user’s particular environment. To the extent permitted by law, MICROSOFT MAKES NO WARRANTY OF ANY KIND, DISCLAIMS ALL EXPRESS, IMPLIED AND STATUTORY WARRANTIES, AND ASSUMES NO LIABILITY TO YOU FOR ANY DAMAGES OF ANY TYPE IN CONNECTION WITH THESE MATERIALS OR ANY INTELLECTUAL PROPERTY IN THEM.

Microsoft may have patents, patent applications, trademarks, or other intellectual property rights covering subject matter within this documentation. Except as provided in a separate agreement from Microsoft, your use of this document does not give you any license to these patents, trademarks or other intellectual property.

Information in this document, including URL and other Internet Web site references, is subject to change without notice. Unless otherwise noted, the example companies, organizations, products, domain names, e-mail addresses, logos, people, places and events depicted herein are fictitious.

Microsoft, Active Directory, Microsoft Dynamics, and Windows Live are either registered trademarks or trademarks of Microsoft Corporation in the United States and/or other countries.

The names of actual companies and products mentioned herein may be the trademarks of their respective owners.

You have no obligation to give Microsoft any suggestions, comments or other feedback ("Feedback") relating to the documentation. However, if you do provide any Feedback to Microsoft then you provide to Microsoft, without charge, the right to use, share and commercialize your Feedback in any way and for any purpose. You also give to third parties, without charge, any patent rights needed for their products, technologies and services to use or interface with any specific parts of a Microsoft software or service that includes the Feedback. You will not give Feedback that is subject to a license that requires Microsoft to license its software or documentation to third parties because we include your Feedback in them.

Contents

[Introduction 1](#_Toc227658545)

[Intended Audience 1](#_Toc227658546)

[About MOF 4.0 1](#_Toc227658547)

[About MOF Companion Guides 1](#_Toc227658548)

[Goals of the Software-plus-Services Companion Guide 1](#_Toc227658549)

[Not in Scope 2](#_Toc227658550)

[Where This Guide Fits Within MOF 3](#_Toc227658551)

[Feedback 4](#_Toc227658552)

[Software-plus-Services: an Overview 5](#_Toc227658553)

[Traditional Software vs. Software-plus-Services 5](#_Toc227658554)

[Software-plus-Services in More Depth 6](#_Toc227658555)

[Key Benefits and Trade-offs of Using Software-plus-Services 7](#_Toc227658556)

[Step 1: Identify Software-plus-Services Opportunities 8](#_Toc227658557)

[Determine Business and Financial Implications 8](#_Toc227658558)

[Activities and Considerations 11](#_Toc227658559)

[Step 2: Determine the Impact of Software-plus-Services on the Business 14](#_Toc227658560)

[Identify Business Requirements and Constraints 14](#_Toc227658561)

[Identify Technical Requirements and Constraints 15](#_Toc227658562)

[Determine How Reliability and Company Policies Will Be Affected 15](#_Toc227658563)

[Activities and Considerations 17](#_Toc227658564)

[Step 3: Select a Service Provider 19](#_Toc227658565)

[Determine Provider Viability from a Business Perspective 19](#_Toc227658566)

[Determine Provider Viability from a Technical Perspective 20](#_Toc227658567)

[Activities and Considerations 22](#_Toc227658568)

[Next Step: Prepare for the New Service 24](#_Toc227658569)

[Appendix A: Key Roles 30](#_Toc227658570)

[Management Accountability 30](#_Toc227658571)

[Architecture Accountability 31](#_Toc227658572)

[Service Accountability 31](#_Toc227658573)

[Appendix B: Key Terms 33](#_Toc227658574)

[Appendix C: Software-plus-Services Models 34](#_Toc227658575)

[Technical Considerations That Define an Application’s Maturity Level 34](#_Toc227658576)

[Technical Considerations for Examining a Multi-Tenant Model 36](#_Toc227658577)

[Acknowledgments 38](#_Toc227658578)

# **Introduction**

This guide takes readers through the process of researching and planning the implementation of a software-plus-services strategy. Initial steps focus on identifying a delivery model that best suits the individual needs of the business; subsequent sections concentrate on finding a qualified service provider, then planning for the implementation itself. This guidance uses Microsoft® Operations Framework (MOF) 4.0 processes to identify the right services, the pertinent considerations, and the initial preparations for cost-effective implementation and operations.

## Intended Audience

This guide is intended for use by business decision makers and the IT pros who will be directly involved in or affected by the implementation of a software-plus-services strategy.

## About MOF 4.0

Microsoft Operations Framework (MOF) 4.0 is concise guidance that helps IT improve service quality while reducing costs, managing risks, and strengthening compliance. MOF defines the core processes, activities, and accountabilities required to plan, deliver, operate, and manage IT services throughout their lifecycle. The MOF guidance encompasses all of the activities and processes involved in managing an IT service: its conception, development, operation, maintenance, and—ultimately—its retirement.

MOF organizes IT activities and processes into service management functions (SMFs), which provide detailed processes and outcomes related to a series of IT disciplines. Each SMF is anchored within a lifecycle phase and contains a unique set of goals and outcomes that support the objectives of that phase. For more information about SMFs, visit [www.microsoft.com/mof](http://www.microsoft.com/mof).

## About MOF Companion Guides

MOF companion guides are intended to help business and technical decision makers and IT pros perform IT-related activities effectively and cost-efficiently. Each guide focuses on a specific activity and applies MOF 4.0 principles. Each guide topic is associated with a particular MOF phase and SMF; readers are encouraged to familiarize themselves with the relevant core MOF material before using the guides.

# **Goals of the Software-plus-Services Companion Guide**

This guide helps answer the following questions:

* Is a software-plus-services strategy right for our business?
* What types of software-plus-services offerings make sense for our organization’s needs?
* How do we go about selecting a service provider?
* After we have selected the plan and the provider, how do we prepare for implementation?

The following table describes the outcomes and specific measures of this guide.

Table 1. Outcomes and Measures of This Guide

| **Outcome** | **Measures** |
| --- | --- |
| An understanding of the potential opportunities, benefits, and risks of adopting the software-plus-services delivery model. | * Identification of key business drivers, objectives, constraints, and other considerations that can be used in developing a business case. |
| A decision to pursue evaluation of an outsourced software-plus-services delivery solution versus maintaining the processes on-premises. | * Business case that outlines the potential benefits, service delivery costs, trade-offs, risks, and values that can be realized. * Supporting rationale and financial justification that articulates the alignment/linkages between business stakeholders’ needs, IT’s software delivery sourcing strategy, and assumptions factored into the business case. |
| A clear set of vendor-evaluation criteria for selecting a software-plus-services provider. | * A list of key questions to ask potential vendors about their service model and capabilities for delivering software-plus-services to enterprise customers. * Vendor-scoring sheets completed across all service delivery and support functions, business case analyses conducted, and vendor references from existing clients checked. * Control matrices used to identify and prioritize potential risks are supported by recommendations for mitigating those risks. |
| Vendor contract preparations and negotiations. | * Due diligence activities completed before executing any contracts. * Financial linkages between vendor payments and penalties established, and the software-plus-services provider’s adherence to the service level agreement (SLA) performance and contract requirements built into the agreements. * A well-defined governance model and vendor management process related to the outsourced software-plus-services delivery project. |

## Not in Scope

Other guides are available from Microsoft that provide additional, more detailed attention to the adoption of specific services. Notably, after the business decision maker targets a specific software-plus-services solution (for example, Microsoft Exchange Online), the technical decision maker can turn to a specific [Infrastructure Planning and Design](http://www.microsoft.com/downloads/details.aspx?FamilyId=AD3921FB-8224-4681-9064-075FDF042B0C&SAMI_CAMPAIGN_NAME=IPDBOSGRTM111308_IPDDL&displaylang=en) (IPD) guide for further assistance.

## Where This Guide Fits Within MOF

The core MOF 4.0 content (found at [www.microsoft.com/mof](http://www.microsoft.com/mof)) discusses the complete IT service lifecycle and provides guidance that is practical and relevant for all IT activities. This software-plus-services companion guide focuses on how the adoption of software-plus-services relates to a specific component of the MOF IT service lifecycle: the Plan Phase.

### Plan Phase

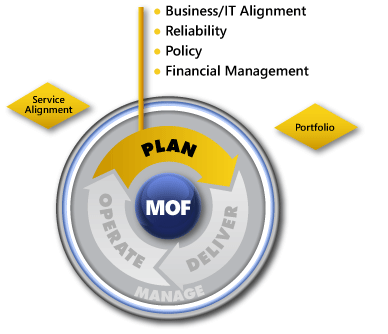


Figure 1. The Plan Phase of MOF

The goal of the Plan Phase is to aid the organization in continually planning for and optimizing its IT service strategy. Its deliverable is a strategy that helps produce services that are valuable and compelling, predictable and reliable, compliant with policies and directives, and adaptable to the changing needs of the business.

Table 2 lists the SMFs included in the Plan Phase.

Table 2. SMFs in the Plan Phase

| **SMF** | **Purpose** |
| --- | --- |
| Business/IT Alignment | * Helps better align business and IT strategy to ensure that IT services provide business value. * Provides an understanding of the fundamental process steps involved and describes the context of aligning business and IT goals, developing an IT service strategy, identifying an IT portfolio of work, and establishing methods of keeping business and IT aligned. |
| Reliability | * Helps IT understand, set targets, and measure service reliability. |
| Policy | * Helps the IT organization determine the areas that require policy creation. * Provides a method for policy creation, validation, publishing, enforcement, evaluation, review, and maintenance. * Helps the IT organization remain in compliance with directives. |
| Financial Management | * Provides an understanding of the fundamental processes and activities involved and describes the context of financial management in terms of risk management and value realization. * Addresses how to establish service requirements and plan budget, manage finances, and perform IT accounting and reporting. |

The following graphic illustrates the interrelationship among the Plan Phase SMFs.

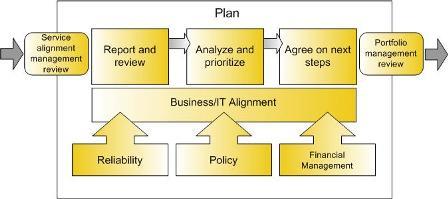
[](http://sharepoint/sites/SATProjects/All/MOF4/MOF%20%20RAW%20Images/fig%202%20plan%20overview%20workflow.jpg)

Figure 2. The interrelationship of the Plan Phase SMFs

## Feedback

Please direct questions and comments about this guide to [mof@microsoft.com](mailto:mof@microsoft.com?subject=MOF%20Companion%20Guide:%20Software-plus-Services).

# **Software-plus-Services: an Overview**

Finding the right solutions to meet business needs gets more complex as the ecosystem of business and technology becomes more diverse. This guide helps you address the questions that must be answered *before* you take advantage of changes in technology and services.

When considering a delivery system for software, businesses have several models from which to choose: on-premises, “cloud”-based (IT-related services provided over the Internet, also known as software as a service, or SaaS), and a “hybrid” combination that Microsoft calls software-*plus*-services. Software-plus-services is an industry shift that combines Internet services with complementary traditional software that runs within the enterprise to deliver more compelling business opportunities and service delivery options.

A hybrid delivery model allows businesses the freedom to choose software services based on their IT strategies and specific needs. The services included in the software-plus-services approach are integrated together and aggregated using a service-oriented architecture approach.

From a business manager’s perspective, software-plus-services can be the best of both worlds: The business benefit can be realized at a cost that is proportional to use (or even free), with no additional or up-front capital investment in IT resources. Although the business is best placed to determine how well the service solves the business problem, unless IT is included in the discussion, many of the wider implications for enterprise IT—the hidden costs of software services adoption—will be missed.

Business leaders and decision makers who are considering software-plus-services need to evaluate the benefits against their business’s requirements, which include service level guarantees, ownership of data, identity management, security, privacy, governance, and compliance.

## Traditional Software vs. Software-plus-Services

Software-plus-services represents a major shift in the way individuals and businesses use software. Users will notice changes in the following areas:

* **User experience.** The traditional software end-user experience centered on a single device: the computer. Software-plus-services supports a large array of items, including browsers and devices of all types, and the user can ideally expect a consistent, seamless experience across all those devices.
* **Delivery.** With software-plus-services, businesses can choose how they get their software: delivered in a box, hosted by a third party, or delivered as an online service.
* **Federation.** The federated business world is a world in which a business doesn’t own and operate every component—rather like the Internet itself. Federation provides flexibility and agility—and therefore, business benefits—from loosely coupled processes. Due to the shared and somewhat open nature of federation, responsibilities for managing and administration must be coordinated, not centrally provided. For example, identity and access rights will be burdensome if agreements and role-based models are not shared across federating partners.
* **Composition.** The federated business world also ushers in a new model of development: Composition is the ability to build solutions that span a variety of assets not owned and operated by a single enterprise.
* **Business models.** New economic models in the software-plus-services world include transactions, subscriptions, advertisements, and micro-transactions. The idea of customers buying a shrink-wrapped box of software, installing and configuring it, and then running it until a newer version is available is rapidly becoming outdated. Instead, acquiring new software and services becomes a more complex and rapidly changing interaction. The range of IT connection points and transaction types are broadening.

## Software-plus-Services in More Depth

Software-plus-services provides more than just a third hosting option (in addition to on-premises and cloud-based); it also affords businesses the ability to host only certain portions of an application or service. In the software-plus-services world, services are building blocks, attached or finished.

Building block services provide low-level capabilities that developers can leverage when building a composite application. These services exist in the cloud.

Attached services provide a higher level of functionality than building block services. Attached services are most useful when attached to some underlying software or service asset. Enterprise and ISV applications leverage attached services in order to add functionality.

Finished services are analogous to full applications, delivered as services over the Internet using the SaaS model. The Software-plus-services approach can use all of these to deliver applications for the organization.

|  |  |  |  |
| --- | --- | --- | --- |
|  | | | |
| **On-premises** |  |  | **SaaS** |
| **Traditional Software** | **Building Block Services** | **Attached Services** | **Finished Services** |
| Installed and operated in-house | Services from which applications can be built | Services with self-contained functionality | Applications delivered via SaaS |
| Targeted to end users | Targeted to developers | Targeted to developers and infrastructure staff | Targeted to end users |
| Example: Microsoft Exchange | Example: Windows Live™ APIs | Example:  Exchange Hosted Services | Example:  Microsoft Dynamics® CRM Online |
|  | ***Software-plus-Services*** | |  |
|  | | | |
| **Self-hosted** |  | **Hosted by service provider** | |

Figure 3. Key characteristics and drivers of hosting choices

## Key Benefits and Trade-offs of Using Software-plus-Services

Adopting a software-plus-services delivery method can result in the following financial and logistical benefits:

* **Reduced capital expenditure**. Capital expenditures required to support peak-level or infrequent intensive computing tasks are reduced with the use of a hosted service provider.
* **Transfer of costs to provider**. Costs related to changing technology, skilled IT staff, and risk management (associated with performance, reliability, and scale) are transferred to the hosted service provider, resulting in a lower barrier to entry for the enterprise.
* **Transfer of responsibilities to provider**. Service delivery and support related to the cloud-based services become the responsibility of the service provider.
* **Reduced effort to support applications on-site**. A portion of service infrastructure is owned by the hosted service provider; as a result there is a reduction in the overall effort focused on businesses to install and run applications in their corporate infrastructures. For services based in the cloud, service delivery, software versioning, software maintenance, and ongoing operations and support are the responsibility of the provider. IT continues to control and has responsibility for software maintenance, ongoing operations, and support of the traditional software or on-premises applications installed in the corporate infrastructure.
* **Increased agility for the enterprise**. Resources are freed to concentrate on business-forward projects, rather than infrastructure support and maintenance. Although reduced costs are an important attribute of this delivery method, the true benefit of software-plus-services is increased agility for the enterprise.

However, there are trade-offs as well. Consider the following issues:

* **Integration.** You will need to address issues of integrating your on-premises systems, including customer service, support and operations management, with those of the service provider.
* **Loss of control**. The service provider’s hosting infrastructure will support multiple customers, not just yours. Consequently, the service provider must manage its internal operational environment, including factors such as security, data leak protection, identity and access management, on-boarding and provisioning new tenants—as well as new ISVs and partners, customer service and support, operations management, internal compliance controls, and lifecycle management.
* **You must thoroughly vet the provider’s infrastructure**. A service provider’s IT infrastructure must provide the availability and reliability that your organization requires. Depending on your risk tolerance, you may need to conduct additional due diligence to ensure that the hosting provider has the equipment, operating experience, and financial maturity required to meet your needs.

The goal of the software-plus-services strategy is to empower customers and partners with richer applications, more choices, and greater opportunity through a combination of on-premises software, partner-hosted software, and Microsoft-hosted software. How software is consumed will ultimately be decided by enterprise business decision makers.

Software-plus-services brings together a set of both on-premises and service-based IT capabilities, enabling enterprise IT to optimize its portfolio of software and services across the infrastructure. This approach helps a business determine its IT investment roadmap within the context of accelerated adoption of services, discover flexibility in delivery options, increase ways to maximize resource usage, decrease total cost of ownership (TCO), and increase value.

# **Step 1: Identify Software-plus-Services Opportunities**

Prior to investing time and capital in researching a software-plus-services strategy, it’s crucial to confirm that your IT group is aligned with the business as a whole. Strategic decisions that directly affect IT, but that are made without the input of IT, are guaranteed to be problematic. However, a lack of communication can reach both ways. If IT decides to implement a certain product or service without communicating with the business first, it might find that that decision doesn’t support the company’s strategic goals. In either case, lack of alignment can result in lost resources, downtime, and antipathy between IT and the business.

Finding ways to effectively and efficiently move the business closer to achieving its goals is an ongoing effort that is described and supported in the [Plan Phase](http://technet.microsoft.com/en-us/library/cc506051.aspx) of MOF.

To ensure that IT and business decision makers are in agreement before investigating software-plus-services, see the MOF 4.0 [Plan Phase Overview](http://technet.microsoft.com/en-us/library/cc543274.aspx) and the [Business/IT Alignment SMF](http://technet.microsoft.com/en-us/library/cc543303.aspx). Perform the processes and steps in that SMF and make note of any misalignments that might need to be corrected before adopting a major new IT strategy.

Businesses are clear about what they want from IT: Services that are reliable, compliant, and cost-effective, and that continuously adapt to ever-changing needs. There are several areas that need to be addressed to make sure that there is a clear connection between the strategy and IT’s portfolio of projects and services. Ensure that:

* IT strategy is aligned to an organization’s broader goals and objectives.
* Delivered IT services are effective and efficient in meeting the organization’s needs.
* IT offerings and services are aligned to the business goals.

In the context of software-plus-services, there is a core set of considerations that should be addressed.

Future trends create opportunities for new solutions and services. By evaluating evolving and disruptive technologies on an ongoing basis, the business can gain insight about investment decisions. These trend evaluations can then be compared to what has been determined to be of value to the business. This comparison is addressed in the next section.

## Determine Business and Financial Implications

Decision makers need to determine whether software-plus-services makes financial sense for the business. This involves weighing this kind of strategy against a conventional, on-premises software model. The analysis should compare the cost drivers from both models, including up-front capital expenditures for hardware, storage, and software licensing; IT personnel costs for installation and deployment; ongoing operations, training, upgrades, maintenance, and support; and ongoing infrastructure costs associated with power consumption, network server capacity increases, and multiple redundant systems.

To complete this analysis efficiently and cost-effectively, decision makers can use the guidance and concepts found in the MOF 4.0 [Financial Management SMF](http://technet.microsoft.com/en-us/library/cc543324.aspx).

Software-plus-services allows areas of software applications and IT functions that support important business processes to be turned over to a third-party service provider. In doing so, the organization can receive many benefits including reduced time to deployment. For example, on-demand business functionality, physical hosting infrastructure, technical support, and operational support services are included in the standard service contract offered by the online service provider.

By leveraging the hybrid delivery model, the enterprise has the flexibility to choose how and where the new business solution is deployed. Critical business data can remain on the internal network, controlled and managed by enterprise IT, and the business application (or certain business functions) can be managed and operated by the online service provider without the enterprise incurring the associated expenses. Although the solution’s installation, acceptance testing, and operation will involve project ramp-up time and a learning curve, the business gets the benefit of a faster time to market compared with an internally developed solution.

Developing the solution internally offers the greatest capability in satisfying corporate standards for security, privacy, usability, localization, and globalization, but also requires internal resources. Additionally, the time-to-market benefit will not be realized until the solution has been developed.

### Financial Considerations for On-Premises Delivery Model

Traditional, on-premises software applications consist of up-front licensing costs, annual renewal for upgrades, and vendor support that is internally deployed at the customer’s location. Costs are divided into three broad categories:

* **Software**. This budget pays for line-of-business (LOB) applications and customizations, including annual licensing, maintenance, and software upgrade fees. Licensing costs are typically based on factors that do not directly correspond to usage, such as server type and number of CPUs. The software model is an isolated, single-tenant application, accessed only by the users of a single customer group.
* **Hardware and infrastructure components**. This budget pays for network and storage equipment, special purpose appliances, servers, desktop computers, mobile devices, network access and bandwidth, and data center facility charges.
* **IT personnel, services, and tools**. This budget pays for ongoing IT people resources, administrative and support tools, skills training, and other IT-related resources required for deployment, maintenance, integration, monitoring, and upgrading. This category also includes external supplier/partner communities, such as consultants and vendor representatives that ensure the continued operation and availability of the system.

The business is responsible for software maintenance, application monitoring, purchase and installation of necessary server hardware, network equipment, and appliances.

The initial cost of new hardware purchases and each license is usually well known, but the ongoing personnel costs (administrative and support) can be more difficult to predict reliably. Ongoing personnel costs for LOB applications are influenced by a number of unpredictable factors (such as unplanned outages) and other hidden costs.

Another challenge in the IT budget process is providing a reliable hardware infrastructure that can easily scale to accommodate usage and growth demands cost-effectively. The purchase of additional servers to maintain availability and reliability of the service also increases up-front capital cost, but failure to make this financial commitment could result in downtime and a loss in end-user productivity.

Likewise, building excess capacity up-front to handle periodic spikes in workload consumption, peak computing periods, and scaling requirements for special events will also increase deployment costs and capital expenditures for hardware, but may result in underutilized computing resources during non-peak periods.

### Financial Considerations for Software-plus-Services Delivery Model

With a software-plus-services model, a business subscribes to a software application without having to provide administrative and support resources or to maintain the underlying server and network infrastructure. The cost breaks down as follows:

* **Initial setup.** These are one-time fees related to implementing the service.
* **Subscription fee.** This is a predictable, recurring fee paid on a monthly basis over the duration of the contract term. It is usually directly aligned with some form of usage pattern—for example, number of seats, named users, and transactions. The subscription fees include the costs of the software, underlying facilities and technology infrastructure, and the IT people services required to operate and maintain the environment.
* **Migration and integration costs.** You might need to alter the service provider’s offerings to meet your specific needs, or you may incur costs for reengineering an application in a service provider’s environment to meet its hosting and monitoring requirements.
* **Termination fees.** Contracts should address what the termination of the relationship will involve, including data and archiving issues, extra fees (if any), support during testing, parallel testing, and cutover periods.

### Reducing the Total Cost of Ownership

The organization must decide whether adopting a software-plus-services delivery approach makes financial sense—whether it delivers the expected values.

Combined with the scalability factor of multi-tenant–efficient applications (see the Appendix), cloud-based infrastructures make it possible for the software-plus-services provider to deliver higher-quality service offerings at a lower software delivery cost as more customers are added. These benefits include:

* Advanced access to skilled/specialized development resources and IT operational expertise without increasing the internal staffing levels, recruitment, and ongoing retention costs.
* Control of the spending associated with non-discretionary costs—for example, third-party licenses, annual support and maintenance agreements, fees associated with externally hosted or managed solutions, and technology upgrades.
* Management of business risk with minimal or no capital investments in software application development or licensed technology platforms.

### Speed to Market

Software-plus-services is a compelling option for organizations looking to gain immediate access to business-specific services—that is, to leverage capabilities that are readily available on the Internet or which can be quickly deployed using a combination of on-premises software and software delivered through a software services provider. Internal IT projects can take months and significant investment to provide similar benefits.

Software-plus-services can also provide a way to leverage new business capabilities that can’t be easily implemented within the enterprise—functionality the business wants, available on-demand. Examples of this kind of service include hosted messaging and online conferencing and collaboration services.

The agility and simplicity inherent in a software-plus-services delivery approach can result in many cost-saving and time-saving advantages. It can allow businesses to:

* Reduce time spent on application deployments—that is, shorten the time periods to install and implement new software applications and upgrades to the latest technology versions.
* Reduce time for IT to respond to changing business demands.
* Increase employee productivity based upon the ability to configure or customize the solution to meet the workflow needs of different user profiles.
* Leverage a technology solution approach that can easily fit into or complement the existing corporate technology infrastructure.
* Significantly influence the longer-term viability of the solution without the associated dependencies and ongoing maintenance and upgrades usually experienced with maintaining on-premises software.
* Directly control or influence the ongoing operational success of the project effort, such as customer satisfaction, service levels, performance, and feature enhancements.

### Business Constraints

At the same time that business drivers are being evaluated, it is important to attend to factors that represent limits to what the business can do.

Decision makers must determine whether the service they’re seeking has any regulatory, security, or privacy considerations that might not be easily addressed if the solution is implemented externally—that is, outside the corporate firewalls. Likewise, certain application functionality may also be deemed too high of a security risk to be deployed outside the internal corporate network.

Conversely, there might be special regulatory, healthcare, or international compliance standards related to security or personally identifiable information (PII) data that can be met more easily using an online service provider.

## Activities and Considerations

The following table summarizes the activities and considerations involved in Step 1.

Table 3. Activities and Considerations for Step 1

| **Activity** | **Considerations** |
| --- | --- |
| Identify areas where software-plus-services can add value | **Key questions:**   * Do any current technology trends give the organization an opportunity to improve services? * Can any new third-party offerings reduce the burden on the organization (for example, message filtering, network management, or help desk services)? * Are some of the services that are traditionally provided or built in-house available as online services (for example, hosted messaging services)? |
| Determine business priorities and financial implications | **Key questions:**   * What is the mission of the organization? Does it differentiate itself by having innovative products, outstanding customer connections, or optimized cost structures? What services would improve realization of the business mission or enhance its differentiation? * What are the organization’s goals? What IT service goals demonstrate direct support of business goals? * Which IT and business representatives should be accountable for the alignment of goals with the adoption of software-plus-services? * How will the business keep IT informed of changes to organizational strategy, plans, and regulatory requirements? * How should the organization request new enhancements to services? * Have the direct, indirect, and hidden costs been identified? * What are the implications for internal resources and compliance issues? * What are the associated risks? * Which key business drivers will be used to calculate the TCO? Compare them using the various approaches to software delivery (on-premises, hosted software services, or hybrid). * Is there benchmark data that defines the cost-per-transaction and/or processing times associated with executing the business functions internally? Will this benchmark data be included in the return on investment (ROI) calculations? * Will the new solution replace an existing system? Are there cost reduction goals that target costs associated with software licensing, third-party maintenance, and ongoing vendor management? * Are there underlying business requirements to lower internal IT resource allocations or annual costs associated with administration, support, and ongoing system maintenance? |
| **Inputs:**   * Commitment and engagement of senior business and IT leadership to define strategic direction and expected impacts of adopting software-plus-services. * Balanced scorecard, business plans, shareholder reports, and strategic roadmaps. * An understanding of what business processes might be common among similar business types (such as retail, manufacturing, or those in the public sector) and expectations for how such common processes would be affected by adopting software-plus-services. * An understanding of which processes or functions deliver differentiation or a unique added value and whether these processes are appropriate for the software-plus-services model. |
| **Outputs:**   * A vision statement of the IT organization’s software-plus-services objectives as well as roles of business and IT partners. * A set of IT high-level goals and objectives that will guide software-plus-services decision making. |
| **Best practice:**   * Create a strategy group that includes senior management from both IT and the business. Include individuals who have a forward-looking view, are respected, have operational understanding, and are accountable to IT. |

# Step 2: Determine the Impact of Software-plus-Services on the Business

Each of the three delivery models (on-premises, externally hosted, and hybrid) offers its own set of benefits and trade-offs that make it an appropriate model to follow in some cases and not in others. This decision is determined by a number of business and technical considerations, as well as the effects each approach has on reliability and company policies. When these factors have been considered, the business can start identifying next steps toward implementing the service.

## Identify Business Requirements and Constraints

Certain requirements and characteristics can significantly affect the type of solution the business chooses and should be thoughtfully considered. Decision makers must consider the following questions:

* **Will the new solution collect, store, or contain interfaces that require access to highly sensitive Personally Identifiable Information (HSPII) data assets**? Examples include government-provisioned identification credentials, financial transaction authorization data, financial profiles, or electronically protected health information.
* **Would disclosure of stored data assets cause considerable loss or liability**? This kind of data includes PII that is not considered highly sensitive. Again, systems supporting this type of data may be better suited for internal implementations and should include oversight from corporate security, audit, and legal to ensure adequate governance and compliance is maintained.
* **Will a governance organization provide oversight to ensure compliance?** Examples include specific auditable controls on financial systems, Sarbanes-Oxley (SOX) compliance, and Health Insurance Portability and Accountability Act (HIPAA) compliance. The frequency and timing of audits are driven by assessing the impact of the risk to the enterprise. Environments that involve high-risk impacts may be better suited for on-premises only implementations.

Lower-risk situations can be deployed using a hybrid delivery model or be externally hosted, depending upon the level of control needed and the methods required to indicate compliance. For example, if the new system becomes in-scope for SOX compliance, there are additional requirements for testing controls and getting appropriate sign-offs. If the hybrid or externally hosted system contains PII data, the data protection requirements may require attestation of compliance by a corporate or regulatory body.

* **Will direct control over daily operations or processing be required by enterprise IT?** Internally hosted systems provide enterprise IT with greater control over the daily operations and processing environment when compared to externally hosted systems managed by a third-party service provider.
* **Will there be a high degree of internal IT control?** Internally developed systems can be more easily extended to take advantage of existing internal infrastructure, as well as the mechanisms and controls used for integration with other internal systems (such as reporting tools).

## Identify Technical Requirements and Constraints

Similarly, technical requirements and characteristics can significantly affect the type of solution the business chooses. Decision makers must consider the following questions:

* **Will integration be limited due to the external segregation of data stores outside the corporate firewalls?** The segregation of data may not be allowed due to corporate security and privacy policies that enforce stringent data protection requirements and controls on how PII data is accessed and shared externally. Deploying the solution internally will enable the most flexibility in consuming feed data and integration options with internal systems, but the software-plus-services model offers an additional level of flexibility regarding where to run the application and where to store the data.
* **Will the new system involve direct consumption of enterprise resource planning (ERP) feed data**? If the new system is externally hosted, the ability to integrate with other internal systems at the data layer may be limited due to the structure of the data and how it can be securely accessed or shared outside company firewalls. Deploying the solution internally will enable the most flexibility to consume ERP feed data and integrate with internal systems.
* **Will there be a need for real-time data exchanges with one or more internal systems**? If the new system is externally hosted, the ability to perform near real-time data exchanges will be directly dependent on the availability and performance of the network transport interface as well as the methods applied to transfer the data and enforce security rules. These issues will need to be incorporated into the overall architecture of the software-plus-services model.
* **Will there be a need for frequent updates or code changes**? Applications designed for the software-plus-services environment can be configured to each customer’s unique requirements. Applications that require frequent, non-configurable changes or customization may be better suited for a dedicated hosted solution or on-premises operations.

## Determine How Reliability and Company Policies Will Be Affected

Decision makers need to identify and prioritize risks related to adopting the software-plus-services model and to provide recommendations to mitigate those risks. To complete this work efficiently and cost-effectively, decision makers can use the guidance and concepts found in the MOF 4.0 [Policy](http://technet.microsoft.com/en-us/library/cc543348.aspx) and [Reliability](http://technet.microsoft.com/en-us/library/cc506069.aspx) SMFs. In particular, policies should be reviewed to ascertain that they appropriately address service providers. Employees should have clear directions for how to engage and interact with service providers, and they must also understand related data governance implications because the data is likely to be transported beyond the organization’s firewall. The service provider should be made aware of policies that will affect them as well. And management needs to determine that appropriate internal controls are in place to support policy objectives and compliance.

### Risk Tolerance

When a business is considering an arrangement that requires corporate data exchange with an external service provider, several factors will ultimately drive the overall approach. These include:

* Available data transport connectivity options.
* Data exchange formats and encryption standards.
* Associated data management and data leak protection services required by corporate policy to safeguard the organization’s data.

Organizations that have very strict information security policies with regard to the external distribution and storage of confidential information should ascertain whether the software-plus-services provider has implemented vigorous information security and change control policies, complemented by a security posture that provides multi-layered defense within the application.

The architecture of the software-plus-services provider’s environment needs to have multiple built-in defense levels that complement one another to provide data protection in different ways, under different circumstances, against both internal and external threats. Security policies and authorization controls on the databases also need to ensure that each organization’s data is kept separate and remains isolated to prevent any customer from accidentally or maliciously accessing other customers’ data.

For some LOB applications (those that rely on highly sensitive data), on-premises applications may be preferred. Keeping the application in-house will, ultimately, help the business maintain the highest degree of control over how data is exchanged between other internally managed systems.

### Data Management Requirements

When considering a provider’s data management capability, the business must verify that the appropriate mechanisms are in place to validate and track the successful or unsuccessful completion of any processing associated with the feed data, transactions, or files sent over the interface. Because of the critical nature of data recovery, the business should set the same requirements for their side of the interface and should ascertain the cutoff or deadline times for both sides.

The business also needs to ensure that the recovery process addresses synchronization of recovered data stores (following a database restore due to a crash or corruption) and that the process supports the continuance of service following disaster recovery capabilities—that is, the ability to restore the data as well as the interface.

## Activities and Considerations

The following table summarizes the activities and considerations involved in Step 2.

Table 4. Activities and Considerations for Step 2

| **Activity** | **Considerations** |
| --- | --- |
| Identify business requirements and constraints | **Key questions:**   * Is there a cyclical nature to the use of the service? For example, is usage heavy during certain events or times of the year? * Will the new system or solution collect, store, or contain interfaces that require access to HSPII data assets? * Will the new solution store any data assets whose unauthorized disclosure would cause considerable material loss to the company or significant regulatory liability? * Will the new system require frequent or direct oversight and control by the organization to ensure that corporate governance, legal, and/or regulatory compliance is maintained? * Is direct control over the daily operations or processing environment required by enterprise IT due to the complexity of business rules, the nature of the business environment, or company security and privacy concerns? * Will the new system require a high degree of internal IT control due to the level of integration needed with other internal systems or because of special data handling, data manipulation, or batch processing requirements? |
| **Best practice:**   * Consider an internal implementation for systems that support or access HSPII data, and ensure that it includes oversight from corporate security, audit, and legal to ensure adequate governance and compliance. |
| Identify technical requirements and constraints | **Key questions:**   * Would the external segregation of data stores outside the corporate firewalls limit the overall ability of the new system to integrate or share data with other internal business systems—for example, limit reporting or processing capabilities? * Will the new solution be required to directly consume ERP feed data or require a deep level of integration with one or more internal systems? * Will the data processing or reporting requirements of the new solution require near real-time data exchanges with one or more internal systems? * Will frequent updates or code changes to the applications, databases, or system processing environment be required due to business rules complexity or a dynamically changing business environment? |
| **Best practices:**   * Create an infrastructure service map that shows the core technical services that support all other IT services. Core technical services might include directory, messaging and collaboration, file management and backup, network, printing, desktop, and server farms. Other service types could include provisioning, support desk, software update, patching, and compliance reporting. * To ensure that a delivery model meets business functionality and experience requirements, validate that there are high rates of user adoption and usage for all deployments. Users should be able to easily, seamlessly, and consistently share files, templates, and information between different applications designed to work together regardless of the hosting location. |
| Determine how reliability and company policies will be affected | **Key questions:**   * What are the provider’s security and change control policies? * What mechanisms does the provider have in place to track completion of processing? |
| **Best practice:**   * For line-of-business (LOB) applications that involve highly sensitive data, consider an on-premises solution. |

# 

# **Step 3: Select a Service Provider**

The final step in the process is to decide which service provider is the best fit for the business and, after that decision has been made, to prepare for that ongoing relationship from a contractual and reporting standpoint.

## Determine Provider Viability from a Business Perspective

The first inquiry into a provider’s capabilities should concern its commitment to service; data management competency alone doesn’t guarantee a satisfactory relationship between client and provider. During due diligence, look for providers with a clearly articulated SLA, an agreed-upon plan for service migration, and a designated account management team.

Bringing IT infrastructure to a “cloud level” capability requires significant investment by a hosted service provider. Depending on your company’s risk tolerance, additional due diligence may be required, especially if the service provider is new to the market or lacks the experience of operating in a commercial hosting environment.

### Service Transition and Data Migration

During the implementation of the new solution or tool, it is critical to address such tasks as configuring the application, establishing the data exchange interfaces, training the end users, and ensuring that the new service capability can be appropriately linked with other companies’ business processes and services.

Consider a provider who has tools for importing and transferring data as well as an established means of governing the implementation, data migration, end-user training, and transition to the new service. Their methodology should outline the process details associated with the different aspects of implementing a new customer.

The process should address data migration from the current (live) system, data cleansing and normalization, and estimates for project resources and each major milestone.

Verify the details of the methodology to ensure the process outlines a plan that will meet your needs as well as temporarily overlap old and new operations to ensure a smooth transition to the new system or hosted service. You may also need to validate the success of the provider’s approach by contacting existing customers with similar needs.

Be sure to include requirements for open standards in the provider selection criteria, and evaluate deployment needs and their dependencies (for example, desktop application compatibility testing, data center space, purchase of integration products). If client deployment is needed, factor the provider release schedule (updates and new versions) into existing engineering processes and plans for client computers.

### Account Management and Support

If it is determined that an account management plan is required, the customer and the provider need to agree on clearly defined roles and responsibilities. Ascertain whether your organization needs a designated individual or team to handle the day-to-day details of your account, escalation management, and business planning, and determine if there is a need for a technical account manager (TAM) on the provider’s side. Find out how customers are notified and kept informed about critical issues or outages affecting service levels.

Clarify the types of activity reporting and logging the provider has available. Look for dashboard tools that provide business views into overall service health, and ensure that you, as a customer, can configure your preferences for receiving automatic notifications of service-related events. Are customers given a way to track the status of open issues or service-related requests? What service management processes need to be shared between the provider’s and client’s IT organizations? Make sure there is an IT-to-IT support model in place.

### Service Level Agreements

Review the standard service level agreement (SLA) between each potential software-plus-services provider and its customers. Make sure that the provider’s SLA states that financial penalties will apply if it fails to meet the established service requirements and that the SLA addresses operational metrics such as performance and availability. The provider should be willing to show its metrics results from the previous six to twelve months.

What security policies and measures does the provider offer? Make sure the SLA spells out the protections against data loss, and compare these to internal company security policies.

The SLA should also define:

* Who owns the data.
* Customer service or support standards (for example, time to response, problem and issue resolution objectives).
* Data protection safeguards (system and data backup cycles, on-site and off-site data retention, archiving, and accuracy of data-handling logs).
* Recovery time for service restoration if service delivery is affected by a natural or man-made disaster.

Make sure that the service provider has a process in place to track customer satisfaction. They should show you (as a prospective client) endorsements by existing customers. They should also be able to measure how well they deliver the customer-contracted services. Do they measure how well their software solution or service ties to the customers’ business success, or the level of impact it has on the customer’s ability to achieve its business goals?

Compare the provider’s SLA with that of the internal IT organization. How do the two compare with regard to service availability, end-user and business team support, business continuance, and disaster recovery? Ensure that these internal obligations can be met or improved under the software-plus-services provider’s service level guarantee.

Expectations and performance targets for the various data exchange interface points should be clearly defined within the SLA and should continue to be part of the ongoing key performance measures.

## Determine Provider Viability from a Technical Perspective

The purpose of any relationship between a company and an online service provider is the handling of data. Discovering how providers address security, regulatory, and data protection requirements is a crucial exercise. In addition, decision makers must be aware of any external sourcing arrangements the provider has made.

### Identity and Access Management

Ensure that the provider’s identity management component supports delegated administration of identities and access policies to respective administrators at separate levels. The provider’s business and operations staff need permissions to perform privileged tasks such as generating billing reports and starting database backups. On the customer side, application administrators need to assign rights to their internal business users for configuring and accessing application features. Administrator account policies for the software-plus-services application need to be evaluated against the organization’s internal corporate account policies (factors such as password complexity and expiration). Consider placing role-based administration onto your provider selection criteria as means of reducing complexity and cost.

#### Provisioning and De-Provisioning Users

Consider how the provider will extend and enforce policies and processes governing internal user provisioning and de-provisioning. For example, when an employee terminates, many organizations struggle to disable internal accounts in a timely manner. In the case of an externally hosted application or service, the risk of exposure could be significant. Will users in the enterprise need their existing corporate accounts to sign on the hosted applications? Multiple approaches are used by software-plus-services providers to satisfy single sign-on (SSO) requirements. For example, directory synchronization, while straightforward, is not terribly scalable or efficient when synchronization needs to take place between the provider’s identity system and many business’s directories.

Another approach is the use of identity federation solutions that enable enterprise SSO by establishing trust relationships with software-plus-services providers who play the role of a third-party identity provider. Standards-based federated identity solutions, such as Active Directory® Federation Services (AD FS), build on trust relationships and policies that allow identity systems to be loosely coupled and therefore more scalable. Consider placing AD FS support onto your provider selection criteria.

Another consideration is scenarios in which the software-plus-services provider hosts applications, some from different software providers. SSO is needed by the software-plus-services provider to reduce unnecessary intrusions on the user experience and the number of credentials the user and application has to manage. The provider must implement the necessary infrastructure for registering, managing, and validating the identities of the application users. You will need to investigate the use of identity technologies that enable the SSO capabilities that work best for your organizational needs and produce an internal architecture that simplifies identity integration of future software services.

#### Considerations

The business case for funding integration projects should be prioritized on improved security and achieving compliance above cost reduction. Important areas to consider include:

* **Monitoring environment**. Delivering reliable software services requires an operational environment that is equipped with the right tools and processes to monitor and correct errors. What are the software-plus-services providers’ monitoring capabilities for detecting, diagnosing, and correcting operational issues? Do their tools provide the infrastructure to monitor performance, availability, and security requirements defined with the SLA? The performance measure focuses on metrics such as throughput and rate of response; availability tools and procedures manage the service accessibility and up time; and security monitoring helps mitigate the risk. Does the operational data derived from monitoring support SLA reporting? On the customer site, you may need to use lightweight monitoring of the service to validate the SLA measurement.
* **IT support model**. Within the customer organization, help desk procedures for the new software-plus-services application will need to be updated to perform first-line troubleshooting, and escalation processes will need to be defined with the service provider. Review internal help desk SLAs to ensure they can still be in place when the service provider is providing escalated support. Is there a need to establish an IT-to-IT support channel for escalating internal IT issues?

### Regulations and Legal Obligations

Determine whether regulatory compliance requirements extend to the provider and, if so, work out what reports, policies, or accreditations are needed to prove compliance. Consider automating the creation of hybrid internal/provider compliance reports. This may mean that the service provider exposes reporting APIs rather than providing a static report. If data is resident at the service provider, review the contract to ensure you retain data ownership. Make sure you extend the internal data integration architecture to support external service integration, add security accreditations to the provider selection criteria, and create an infrastructure integration architecture to provide a framework for software service integration.

### Data Protection Requirements

Part of due diligence involves discerning how well the provider deals with data protection issues. Thus, decision makers should carefully review a provider’s resilience and disaster recovery policies and procedures. If data is resident at the provider, determine the necessity and possibility of block-level restores (where raw data blocks changed on the disk or volume are backed up incrementally).

## Activities and Considerations

The following table summarizes the activities and considerations involved in Step 3.

Table 5. Activities and Considerations for Step 3

| **Activity** | **Considerations** |
| --- | --- |
| Determine provider viability from a business perspective | **Key questions:**   * Does the provider’s standard SLA address operational metrics such as performance and availability? * Is the provider willing to show its metrics results from the previous six to twelve months? * What security policies and measures does the provider offer? Does it spell out the protections against data loss? How do these policies compare to internal company security policies? * Does the service provider offer self-service tools for provisioning new services, configuring service settings, and managing system health? Do the tools support account administration by non-IT personnel or other users, and can they be configured for self-service? |
| **Best practices:**   * Ensure that the provider’s SLA defines the following: * Ownership of the data. * Customer service or support standards (for example, time to response, problem/issue resolution objectives). * Data protection safeguards (system and data backup cycles, on-site and off-site data retention, archiving, and accuracy of data-handling logs). * Recovery time for service restoration if service delivery is affected by a natural or man-made disaster. * When formulating an account management plan with the provider, ensure that everyone is in agreement about roles and responsibilities. |
| Determine provider viability from a technical perspective | **Key questions:**   * What are the integration requirements? Will the integration be via technology or a human process? * Do firewall rules need to be updated to provide filtering for application data (for example, XML schema validation)? Are firewall rules needed to allow for integration? * Are certificates or implementation of PKI needed to support authentication, encryption, and signing requirements? |
| **Best practices:**   * Consider the cost of implementing a system that enables synchronization between the service provider’s identity management system and the enterprise’s directory system versus the labor costs incurred by the service provider’s help desk for performing the task manually. * Verify that the appropriate notification and recovery processes exist on both interface ends to handle abnormal conditions such as failures in technology, malformed files, and files lost during transit over the Internet. |

# 

# **Next Step: Prepare for the New Service**

After the services have been mapped and a service provider identified, decision makers need to identify and complete any necessary agreements and contractual considerations prior to implementing the new service. This includes detailed considerations of policies, management and reporting processes, work flows and reliability, and the pre-adoption work involved as a service is moved into production, along with operation and support issues. The tasks in Table 6 can help business decision makers:

* Consider the risk implications of expanding IT's software delivery portfolio beyond its firewall.
* Assess how it affects existing IT assets.
* Identify steps to mitigate risks associated with making the transition to a new software delivery model.

The following tasks and best practices are set up as a table to facilitate a clear division of responsibility.

Table 6. Pre-Implementation Tasks

| **Responsible Party** | **Actions** |
| --- | --- |
| IT Governance Decision Maker | * Assign executive sponsor or leadership team with primary responsibility for incorporating software-plus-services delivery into the organization’s IT governance structure. * Set strategic direction and objectives for the managing software-plus-services delivery and relationship between parties. * Align business case, value metrics, and measurements to expected value to be realized. * Develop governance operating model to supply decision-making rights, management reviews, resolution of escalated service issues or conflicts, service performance, and change management to the service requirements. * Determine key roles, responsibilities, and accountabilities between the two organizations. * Assign individuals to roles within IT governance structure.   **Note**   For more information, see the [Governance, Risk, and Compliance SMF](http://technet.microsoft.com/en-us/library/cc531019.aspx). |
| Software-plus-Services Provider Account Manager | * Develop business control process to maintain oversight of software-plus-services vendor performance per criteria defined for governance. * Require the software-plus-services vendor to provide reporting that compares actual service delivery against the applicable SLA or performance metrics. * Establish regular meeting schedules to review service level performance and changes requirements to software-plus-services delivery needs. * Review and monitor delivery of software-plus-services according to contractual terms and conditions. * Assess whether the expected business value sought is being realized. * Monitor stakeholder adoption levels, escalated issue resolutions, and new stakeholder business requirements to ensure the relationship meets expectations. * Establish a feedback loop with software-plus-services vendor to ensure that timely communication of employee issues, system errors, or process problem trends occurs for strategic resolution. * Review internal change requests and analysis documentation and act as a liaison with software-plus-services vendor to secure changes and approvals to the agreement scope. * Consider making software-plus-services vendor management a dedicated account management function rather than a function of another job. Non-dedicated personnel performing vendor management can lead to poor oversight of the vendor’s performance. * Consider assigning multiple layers of account management as part of the governance model. * Develop a plan to rotate account managers periodically to ensure that arm’s-length relationships are maintained. |
| Software-plus-Services Provider IT Operations | * Manage the day-to-day service operations with clearly defined, mature IT control processes. * Maintain a multi-pronged, multi-tiered security approach to ensure customer information/data assets are protected at all times. * Monitor all service level and other performance metrics defined by the SLA. * Provide around-the-clock, in-house technical expertise to support and resolve any internal operational issues or problems affecting service delivery. * Notify customer organization of unresolved or escalated issues or problems affecting software-plus-services vendor’s delivery of software-plus-services. * Monitor all critical components of the applications, databases, and infrastructure-related components to ensure that availability, performance, and capacity expectations for the service are being maintained. * Provide full lifecycle management services including administrative, technical, and engineering resources required to install, maintain, troubleshoot, and operate the offered software systems and services. * Report operational performance showing actual service delivery against the applicable SLA or performance metrics. * Review and implement approved changes requested to the agreement scope. * Facilitate integration with customer’s identity management and access control systems, data exchange interfaces, and compliance reporting. * Participate in [policy and control management reviews](http://technet.microsoft.com/en-us/library/cc506048.aspx) to make sure that compliance objectives are met and that the people in identified roles regularly meet and fulfill their mutual obligations. |
| Decision Maker Focused on Service Level Agreements | * Establish SLAs with the software-plus-services vendor that define the scope of the service guarantees, including metrics used to measure availability, performance, security, and support. * Ensure the SLAs provide sufficient levels of mitigation (for example, data loss protection, backup and recovery services, and disaster recovery capabilities) in even the worst-case scenario. * Ensure the SLAs guarantee a high level of service to your business and define the actions the provider will take—or the compensation it will provide—in the event that it fails to meet these guarantees. * Ensure that the penalties for nonperformance, poor performance, or contract noncompliance are sufficiently significant to motivate the vendor to meet business requirements. * Consider building a bonus structure into the contract to reward the software-plus-services vendor for consistently performing better than what the SLAs require. * Define a consistent and documented process for managing SLA exceptions resulting in service interruptions. * Promote a commitment to quality and the continuous improvement process by conducting adequate due diligence when SLAs are missed: Determine the root cause and perform timely remediation. |
| Business Intelligence/Data Reporting Services | * Determine what reporting services the software-plus-services provider offers and whether they are compatible with your business-intelligence requirements. Because software-plus-services delivery may involve giving up direct control to some level of corporate data, accurate and useful reporting services may be needed across internal and software-plus-services vendor-hosted systems to verify that data is in synch and that interfaces are operating as expected. * Develop a process to identify software-plus-services vendor data errors and to research and resolve possible data integrity issues. |
| Regulatory Compliance Officer | * Develop a process to systematically coordinate policy changes, regulatory compliance procedures, and new initiatives to the software-plus-services vendor and define responsibility for this function in the governance model. * If the software-plus-services vendor has a SAS 70 accreditation, determine if it applies to a Type I or Type II audit. * Review the software-plus-services vendor’s SAS 70 report thoroughly to determine if the processes and controls being tested apply to the data center, infrastructure, or application services provided by the software-plus-services vendor’s offerings under consideration. Be aware that many providers will claim SAS 70 Type II, but it only applies to the hosting infrastructure and not the hosted applications. * Examine the report thoroughly to determine whether the provider can comply with your own policies. |
| Billing and Invoicing Personnel | * Review the software-plus-services vendor’s pricing model for costs or charges related to any initial setup, installation, or implementation fees, monthly recurring subscription or usage fees, and one-time charges (such as customizations or data migrations). * Establish controls and respective ownership around invoice review, reconciliation, authorization, and payments. * Establish a linkage between invoicing and software-plus-services vendor performance, SLA adherence, satisfaction survey results, and contract requirements. * Develop a process whereby the organization’s vendor account management team reviews the invoices for accuracy, validates that SLAs and performance requirements were met, and payments were authorized accordingly. |
| Confidentiality and Privacy Manager | * Regularly assess the software-plus-services vendor’s privacy and confidentiality compliance defined by your internal policies, and assign responsibility for this function in the governance model. * Determine if the software-plus-services provider privacy policies and practices are in compliance with requirements set forth by the U.S. Department of Commerce “Safe Harbor” framework. Certifying to the safe harbor will assure European Union (EU) organizations that your company provides adequate privacy protection for transferring personal data. * Develop a process to manage privacy and confidentiality complaints related to the software-plus-services provider outsourcing arrangement. This should include defining an escalation path and a process to resolve complaints with the software-plus-services vendor. * Verify that the software-plus-services vendor’s employees and subcontractors with access to your organization’s data are required to sign a non-disclosure agreement and undergo background checks as a condition of employment. * Require the software-plus-services vendor to administer privacy and confidentiality training to all employees and subcontractors handling your organization’s personal data. * Limit vendor and subcontractor access to sensitive, employee-personal data, such as government-provisioned identification credentials (for example, social security or driver’s license numbers), financial transaction authorization data (for example, credit card information), financial or medical profiles, and other highly sensitive information where unauthorized disclosure would cause considerable material loss. |
| Employee Liaison | * Define the timing, frequency, and population of employees that will be surveyed, and develop a process to systematically collect and summarize survey results. * Develop a process to confirm that the survey results are reviewed by the appropriate management teams and to address and resolve concerns identified in the surveys in a timely manner. |
| Data-Security Specialist | * Assess your information security policy and data-security needs to ensure the software-plus-services provider has sufficient security measures and data protections in place to meet your corporate standards. * Evaluate the software-plus-services provider’s security measures for its data center, networks, servers, and SaaS application security. Additional considerations for protecting data include firewalls, digital certificates, security scans, vulnerability assessments, and industry-recognized security certifications. * Review the software-plus-services vendor’s backup and data recovery capabilities: frequency and type of backups, off-site storage, retention periods, and archiving services. * Review the software-plus-services vendor’s disaster recovery and business continuity plans and the testing of those plans. * Develop an employee satisfaction survey about the success of the software-plus-services offering. Include questions about the success of the transition process and identify issues and opportunities for improvement. |
| Manager in Charge of Implementation | * Review software-plus-services provider’s implementation methodology and procedures it uses, including any provisions for data-migration and identity integrations required for single sign-on. * Develop a plan for migrating to the new software-plus-services provider. Include a complete list of activities each party is responsible for during the implementation process. * Identify risk factors, risk mitigation strategies, necessary security tasks, data preparation tasks, communication plans, and other measures necessary to minimize disruption to the organization during implementation. * Transfer knowledge through training and other measures to facilitate using the software-plus-services applications. |
| Contracts Manager | * Develop exit plan for migrating data out of the application in the event the contract is terminated prematurely for reason or expires naturally. Ensure the contract language specifically addresses: * Terms and conditions in effect upon termination or expiration of the agreement. * Termination assistance services, fees, charges, or other compensation. * Control and ownership of data throughout the life of the contract. * Source code disposition, including code escrow for any derivative works created during the contract term. * Specifications, documentation, information, and other assistance necessary to enable the organization to receive services from another provider. |

# Appendix A: Key Roles

The primary Team SMF accountability that applies to the software-plus-services companion guide is the Management Accountability. Roles within the Architecture and Service accountabilities are also involved. See the [MOF Team SMF](http://technet.microsoft.com/en-us/library/cc543311.aspx) for more information about role types and accountabilities.

## Management Accountability

The following table lists the role types associated with the Management Accountability, as well as the responsibilities and roles for each role type.

Table A-1. Management Accountability and Its Attendant Role Types

| **Role Type** | **Responsibilities in This Guide** | **Role in This Guide** |
| --- | --- | --- |
| IT Executive Officer | * Sponsors IT software-plus-services initiatives * Uses metrics and benchmarking to evaluate software-plus-services business impacts * Approves structures and overall externalization of IT processes * Owns board and executive relationships * Engages in decision making | * Sponsorship * Owns external software delivery process and IT decision making * Ensures clear ownership and accountability * Clear trends in software-plus-services performance |
| IT Manager | * Identifies and engages appropriate business and IT stakeholders in decision process * Manages risk and IT business value realization dependencies * Owns business/IT relationship * Manages externalization of IT | * Oversees business/IT alignment processes * Drives accurate forecasting of IT resources into the software-plus-services business case |
| IT Finance Manager | * Manages the financial aspect of the software-plus-services business case development | * Ensures IT budget and accounting are accurate and timely * Assesses the software-plus-services vendor’s financial health |
| IT Policy Manager | * Works with business, management, and legal resources to define policy requirements * Responsible for industry regulatory knowledge * Ensures that management decisions are informed by policy and that policy is effectively used across IT | * Provides policy input to decision process—helps ensure that the software-plus-services delivery model adopted can deliver and enforce policies that address business, regulatory, and industry requirements * Identifies criteria required to maintain oversight and governance of vendor performance |
| Business Relationship Manager | * Acts as a link between IT and the business and partners | * Validates that IT understands business requirements * Considers technology opportunities and constraints in business strategy |

## Architecture Accountability

The following table lists the role types associated with the Architecture Accountability, as well as the responsibilities and roles for each role type.

Table A-2. Architecture Accountability and Its Attendant Role Types

| **Role Type** | **Responsibilities in This Guide** | **Role in This Guide** |
| --- | --- | --- |
| Architecture Manager | * Accountable for ensuring creation and maintenance of architecture plan specific to external identity and access management, maintaining security and privacy compliance, and transferring/managing data integrations with external software-plus-services providers | * Uses Governance, Risk, and Compliance (GRC) and Reliability requirements to provide input to evaluation of software-plus-services vendor |
| Architect | * Considers future directions and solutions that simplify identity, access, and data integration of future external software services * Produces an Infrastructure Integration Architecture that provides for integrations with external software-plus-services providers | * Provides deep-level subject matter expertise specific to identity and access management, security and privacy compliance, and transfer/management of data integrations |

## Service Accountability

The following table lists the role types associated with the Service Accountability, as well as the responsibilities and roles for each role type.

Table A-3. Service Accountability and Its Attendant Role Types

| **Role Type** | **Responsibilities in This Guide** | **Role in This Guide** |
| --- | --- | --- |
| Supplier Manager | * Tracks external vendors who provide supporting services and products | * Ensures effective software-plus-services vendor relationships can be established according to policy |
| Account Manager | * Acts as a link between users or customers and the IT organization * Meets with the customer, discusses current issues, and makes sure that expectations are aligned | * Ensures effective customer and user relationships * Drives relationship with the business |
| Service Level Manager | * Accountable for Business/IT alignment * Acts as main interface between the business and the IT service delivery and support organizations * Handles all issues and development in Service Level Management, including development and agreement of SLAs, operating level agreements (OLAs), and underpinning contracts (UCs) * Represents the business, but works with and within the IT organization | * Ensures effective IT service delivery is specified within the software-plus-services provider’s SLAs |
| Customer Service Manager | * Accountable for the end-user support plan and development of the partner support model required for handling and reporting escalated issues | * Ensures effective end-user support is specified within the software-plus-services provider’s SLAs * Ensures internal help desk SLAs can still be met when depending on the software-plus-services provider for support |

# Appendix B: Key Terms

The following table defines key terms used in this guide.

Table B-1. Key Terms

| **Term** | **Definition** |
| --- | --- |
| cloud computing | IT-related services provided over the Internet. |
| hosted service provider | A business that delivers a combination of traditional IT functions, such as infrastructure, applications (software as a service), security, monitoring, storage, Web development, Web site hosting, and e-mail, over the Internet or other wide area networks. A hosted service provider combines the abilities of an application service provider (ASP) and an Internet service provider (ISP). |
| hybrid hosting model | A combination of on-premises and third-party hosted services. |
| multi-tenant applications | Refers to a software architecture design that accommodates more than one customer using a single instance of an application, database, or compute resource. |
| SAS 70 Audit | A widely recognized auditing standard developed by the American Institute of Certified Public Accountants (AICPA). A Type I report describes the service organization's description of controls at a specific point in time (for example, June 30, 2008). A Type II report includes not only the service organization's description of controls, but also detailed testing of the service organization's controls over a minimum six-month period. |
| service level agreement (SLA) | A written agreement that documents required levels of service. The SLA is agreed upon by the IT service provider and the business, or by the IT service provider and a third-party provider. SLAs should list the metrics and measures that define success for both IT and the organization. |
| software as a service (SaaS) | Software deployed as a hosted service and accessed over the Internet. Line-of-business (LOB) services are often large, customizable business solutions aimed at facilitating business processes such as finances, supply-chain management, business productivity services such as e-mail, and customer relations. These services are typically sold to customers on a subscription-basis. |
| software-plus-services | Client-desktop, client-browser, client-device, and server-based applications that consume one or more Internet (cloud) services. |

# **Appendix C: Software-plus-Services Models**

The process of evaluating service providers involves setting expectations and making comparisons. One approach is to evaluate the maturity of service providers in terms of the architectures and management approaches that are the basis of the services they offer. Underlying this approach is assessing how applications are instantiated on the hosting platform, and how the service provider stores and manages the data that passes through these applications. This appendix describes the technical aspects of these application maturity levels.

## Technical Considerations That Define an Application’s Maturity Level

Using the following differentiators as criteria, decision makers can examine a software-plus-services application’s maturity using an architecture model with four distinct levels:

* Level I: Ad hoc/custom
* Level II: Configurable
* Level III: Configurable, multi-tenant efficient
* Level IV: Scalable, configurable, multi-tenant efficient

### Level I: Ad Hoc/Custom

Architecturally, this level is similar to traditionally sold business software. Key attributes include a single-instance, customizable version of the hosted application for each customer organization.

At this level, each instance of the LOB application is hosted in a dedicated or consolidated server hardware infrastructure through the use of virtualization technologies; a single instance running on the physical server is logically operated independently of other instances. The application's code base can be customized and data model extended to meet individual customer needs. Each customer has its own set of databases that remain isolated from data that belongs to other customers, so restoring data from backups in the event of a failure is a relatively simple procedure.

Software delivered at this level has more significant hardware and maintenance requirements and operating costs compared with the other levels, and it offers few of the benefits of a fully mature software-plus-services solution. However, it can provide a viable alternative sourcing model if a business wants to reduce IT infrastructure and operations costs and is willing to pay a “premium” for added security and customizability. For example, customers in fields such as banking or medical records management often have very strong data isolation requirements; they may not even consider an application that does not supply each tenant with its own individual database.

### Level II: Configurable

At the second level of maturity, the service provider hosts a separate instance of the application for each customer (or tenant), but all instances use the same code implementation. In the first level, each instance is individually customized for the tenant. At level II, the software application publisher meets individual customers' needs by providing detailed configuration options (configurable metadata) that allow the customer to change how the application looks and behaves to its users. Despite being identical to one another at the code level, each instance remains wholly isolated from all the others.

Moving to a single code base for all of the service provider’s customers greatly reduces the service and maintenance requirements of a software-plus-services application. Any changes made to the code base can be easily provided to all of the customers at once.

At the data architecture layer, each tenant is given its own separate, isolated database, which makes it easier to extend the application's data model to meet the individual needs of tenants; additionally, restoring a tenant's data from backups in the event of a failure is a relatively simple procedure. Much like the first maturity level, this approach tends to lead to higher equipment costs for the servers as the number of tenants that can be housed on a given database server is limited by the number of separate databases that the server can support.

Similarly to the first maturity level, software delivered at the second level requires that the hosting vendor provide sufficient hardware and storage capacity to meet the scaling requirements for supporting a potentially large number of application and database instances running concurrently.

### Level III: Configurable, Multi-Tenant Efficient

At the third level of maturity, the service provider hosts a *single* instance of the application that serves many individual customers, with configurable metadata providing a unique user experience and feature set for each one.

In this approach, computing resources and application code are generally shared between all the tenants on a server, but the data architecture and degree of isolation can vary significantly, depending on technical and business considerations.

Another approach involves housing multiple tenants in the same database, with each tenant having its own set of tables that are grouped into a schema created specifically for the tenant. The shared database, separate-schemas approach offers a moderate degree of logical data isolation for security-conscious tenants, although not as much as a completely isolated system would; it can support a larger number of tenants per database server.

The separate schema approach is appropriate for applications that use a relatively small number of database tables, approximately 100 or fewer per tenant. This approach can typically accommodate more tenants per server than the separate-database approach can, so the application can be offered at a lower cost, as long as customers will accept having their data co-located with that of other tenants.

A significant drawback of the separate-schema approach is that tenant data is more difficult to restore in the event of a failure. Restoring the entire database would mean overwriting the data of every tenant on the same database with backup data, regardless of whether each one has experienced any loss. Therefore, to restore a single customer's data, the database administrator may have to restore the database to a temporary server, and then import the customer's tables into the production server—a complicated and potentially time-consuming task.

This approach eliminates the need to provide server space for as many instances as the vendor has customers, which allows for much more efficient use of computing resources than the second level and translates directly to lower costs. A significant disadvantage of this approach is that the scalability of the application is limited. Unless partitioning is used to manage database performance, the application can be scaled only by moving it to a more powerful server (scaling up), until diminishing returns make it impossible to add more power cost-effectively.

### Level IV: Scalable, Configurable, Multi-Tenant Efficient

At the fourth and final level of maturity, the software-plus-services provider hosts multiple customers on a load-balanced farm of identical instances, with each customer's data kept separate, and with configurable metadata providing a unique user experience and feature set for each customer. A software-plus-services system is scalable to an arbitrarily large number of customers because the number of servers and instances on the back end can be increased or decreased as necessary to match demand, without requiring additional re-engineering of the application, and changes or fixes can be rolled out to thousands of tenants as easily as to a single tenant.

## Technical Considerations for Examining a Multi-Tenant Model

Decision makers can use the following key attributes and differentiators to examine the implications of several multi-tenant database architectures:

* Dedicated tenant database
* Shared database, fixed extension set
* Shared database, custom extensions

### Dedicated Tenant Database

Each tenant is provided their own individual database, which can be extended as necessary. With this approach, a new standard default database is created for a new tenant as part of the provisioning process, and each tenant has its own set of data that remains logically isolated from data that belongs to all other tenants. Once the new database is created, the tenant is free to modify it as extensively as the application's user interface and program logic allows, potentially creating new fields, new queries, and even new tables and relationships. Metadata associates each database with the correct tenant, and authorization and security policies ensure that each customer's data is kept separate from that of other customers to prevent any tenant from accidentally or maliciously accessing other tenants' data.

Of the available options, this is the simplest arrangement for software publishers to build, and it offers customers the maximum freedom to extend the default data model. This architecture approach is best for an organization that has very strong data isolation requirements (such as banking or medical records management). Operationally, when each tenant has its own database, the software-plus-services provider can restore a single tenant's data from the most recent backup following a relatively simple procedure.

The disadvantage to the approach is that it tends to lead to higher equipment costs on the back end, as the number of tenants that can be housed on a given database server is limited by the number of separate databases the server can support.

### Shared Database, Fixed Extension Set

A single database is shared by all tenants, and includes a preset number of custom fields that tenants can assign and use as desired. Records from different customers are intermingled in a single table, and a TenantID field associates each record with an individual tenant. In addition to the standard set of fields, a number of custom fields are provided; each tenant can choose how to use these fields and how data will be collected for them.

Custom fields can be typed, so the customer can use any available built-in type checking and verification functions that the application and database provide in order to validate the data. Alternatively, the fields can be untyped, so the customer can use them to store any type of data. (The customer can optionally provide its own validation logic in order to prevent users from accidentally entering invalid data).

A shared database carries a much lower cost for providing services than the isolated approach does because it allows a single database engine to support a larger number of customers before partitioning becomes necessary. The biggest disadvantage to this approach is that the extensibility of the data model is limited to the number of custom fields the application provides. If there are too few custom fields, your customers will not be able to use the application effectively; if there are too many, the result is a sparse, wasteful database with many unused fields.

### Shared Database, Custom Extensions

A single database shared by all tenants allows customers to extend the data model arbitrarily, storing custom data as name-value pairs in a separate table. Each customer record that includes custom data is assigned a unique record ID, which matches one or more rows in a separate extension table. For each row in this table, a name-value pair is stored. Each customer can create as many of these name-value pairs as is necessary to meet their business needs. When the application retrieves a customer record, it performs a lookup in the custom data table, selects all rows corresponding to the record ID, and returns them to be treated as ordinary field data. Obviously, data in the custom data table cannot be typed because it is likely to contain data in many different forms for different customers. To work around this limitation, a third column can optionally hold a data-type identifier so that the data can be cast to the appropriate data type once it is retrieved.

This approach makes the data model arbitrarily extensible, while retaining the cost benefits of using a shared database. The main disadvantage is an added level of complexity for such database functions as searching, indexing, querying, and updating records. This is typically the best approach to take if the business anticipates that customers will require a considerable degree of flexibility in extending the default data model, but will not require data isolation.

When developing an extensibility approach for the data model, remember that any extension implemented by a customer will require a corresponding extension to the business logic (so that the application can use the custom data), as well as an extension to the presentation logic (so that users have a way to enter the custom data as input and receive it as output). The configuration interface presented to the customer should therefore provide mechanisms for updating all three, preferably in an integrated fashion.

# **Acknowledgments**

The Microsoft Operations Framework team acknowledges and thanks the people who produced Planning for Software-plus-Services: A MOF Companion Guide. The following people were either directly responsible for or made a substantial contribution to the writing and development of this guide:

Chuck Chemis, *ExamineOPS*

Joe Coulombe, *Microsoft*

Laurie Dunham, Microsoft

Clare Henry, *Microsoft*

Jennifer Kerns, *Wadeware LLC*

Don Lemmex, *Microsoft*

Betsy Norton-Middaugh, *Microsoft*

Jason Osborne, *Microsoft*

Ruth Preston, *Volt*

Pat Rytkonen, Volt