ISPE® | GAMP®

GOOD PRACTICE GUIDE:

Enabling Innovation

Critical Thinking, Agile, IT Service Management

GOOD PRACTICE GUIDE:

Enabling Innovation

Critical Thinking, Agile, IT Service Management

Disclaimer:
The *ISPE GAMP® Good Practice Guide: Enabling Innovation – Critical Thinking, Agile, IT Service Management* seeks to apply *ISPE GAMP® 5* principles and current good practice to these areas to promote innovation and advancement. This Guide is solely created and owned by ISPE. It is not a regulation, standard or regulatory guideline document. ISPE cannot ensure and does not warrant that a system managed in accordance with this Guide will be acceptable to regulatory authorities. Further, this Guide does not replace the need for hiring professional engineers or technicians.

Limitation of Liability
In no event shall ISPE or any of its affiliates, or the officers, directors, employees, members, or agents of each of them, or the authors, be liable for any damages of any kind, including without limitation any special, incidental, indirect, or consequential damages, whether or not advised of the possibility of such damages, and on any theory of liability whatsoever, arising out of or in connection with the use of this information.

All trademarks used are acknowledged.

ISBN 978-1-946964-46-5

Preface

The drive within life sciences to improve patient safety and product quality, and provide value to society, while simultaneously reducing costs requires constant and effective innovation. Operating in a highly regulated sector can result in some practitioners applying unthinking, prescriptive, and rigid approaches that are not commensurate to the needs of the process, the nature of the system, and the real risk to the product and the patient.

This *ISPE GAMP® Good Practice Guide: Enabling Innovation – Critical Thinking, Agile, IT Service Management* discusses three topic areas where regulated companies can apply innovation to meet rapidly changing industry needs. This Guide facilitates the effective and efficient use of valuable resources by the application of appropriate and proportionate practices, so that companies can introduce innovative approaches to reduce risk to patient safety, product quality, and data integrity.

This Guide is aligned with the concepts and framework of the *ISPE GAMP® 5 Guide: A Risk-based Approach to Compliant GxP Computerized Systems.*

Acknowledgements

The Guide was produced by a Task Team led by Heather Watson (GlaxoSmithKline, United Kingdom), Chris Clark (TenTenTen Consulting Limited, United Kingdom), and Sion Wyn (Conformity Ltd., United Kingdom). The work was supported by the ISPE GAMP Community of Practice (CoP).

Core Team

The following individuals took lead roles in the preparation of this document.

Mark Cherry	AstraZeneca	United Kingdom
Chris Reid	Integrity Solutions Limited	United Kingdom
Judith Samardelis	Syneos Health	USA
Lorrie Vuolo-Schuessler	Syneos Health	USA
Charlie Wakeham	Waters Corporation	Australia
Guy Wingate	GlaxoSmithKline	United Kingdom

Contributors

The Leads thank the following individuals for their valuable contribution during the preparation of this Guide.

Karen Ashworth	Karen Ashworth Consulting Ltd.	United Kingdom
Carsten Bierans	Körber Business Area Pharma	Germany
Stephen Ferrell	Compliance Path Ltd.	USA
James Gunning	Johnson Matthey	USA
Frank Henrichmann	QFINITY	Germany
Oliver Herrmann	QFINITY	Germany
Paul James-Martin	Integrity Project Solutions Limited	United Kingdom
Kevin C. Martin	Azzur Group LLC	USA
Andrew McDonagh	Emergn	United Kingdom
Elizabeth McLellan	Suvoda, LLC.	USA
Khaled Moussally	Compliance Group	USA
Ray Murphy	Boston Scientific	USA
Donal O'Brien	Dassault Systèmes	Ireland
Michael Osburn	Cornerstone OnDemand	USA
Randy Perez	Novartis (retired)	USA
Jens Seest	Novartis	Germany
Ken Shitamoto	Gilead Sciences, Inc.	USA
Andy Tyrrell	Cornerstone OnDemand	United Kingdom
Anders Vidstrup	NNIT A/S	Denmark
Mark Walton	Cornerstone OnDemand	United Kingdom
Anette Westphal	Novo Nordisk A/S	Denmark
Christopher White	National Resilience, Inc.	USA

Regulatory Input and Review

Particular thanks go to the following for their review and comments on this Guide:

Arno Terhechte, GMP Inspector	Bezirksregierung Münster (District Gouvernement Münster)	Germany

Special Thanks

The Team Leads would like to give particular thanks to Mike Rutherford (Syneos Health, USA) for his efforts during the preparation of this Guide.

The Leads would also like to thank ISPE for technical writing and editing support by Jeanne Perez (ISPE Guidance Documents Technical Writer/Editor) and production support by Lynda Goldbach (ISPE Publications Manager).

The Team Leads would like to express their grateful thanks to the many individuals and companies from around the world who reviewed and provided comments during the preparation of this Guide; although they are too numerous to list here, their input is greatly appreciated.

Company affiliations are as of the final draft of the Guide.

600 N. Westshore Blvd., Suite 900, Tampa, Florida 33609 USA
Tel: +1-813-960-2105, Fax: +1-813-264-2816

www.ISPE.org

Table of Contents

1 Introduction

The ISPE GAMP® Community of Practice (CoP) [1] promotes the understanding of regulations and use of GxP computerized systems within the pharmaceutical, biopharmaceutical, medical device, and other regulated industries. *ISPE GAMP® 5: A Risk-Based Approach to Compliant GxP Computerized Systems* [2] is the leading international guidance document in this area:

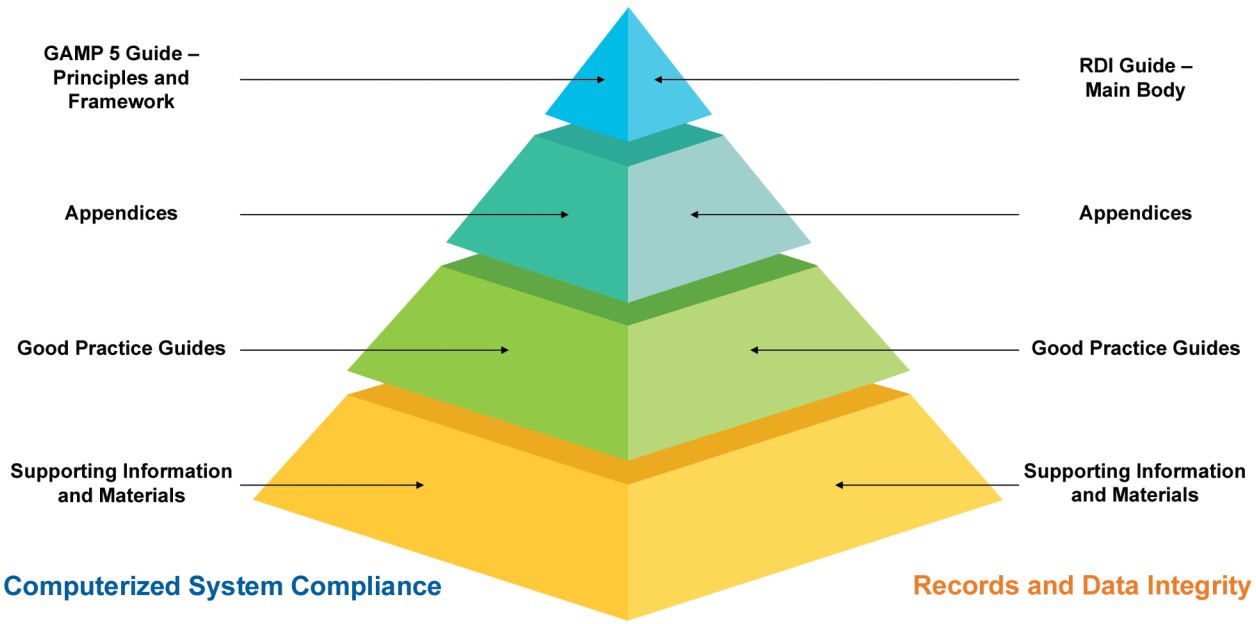
> *"GAMP® guidance aims to achieve computerized systems that are fit for intended use and meet current regulatory requirements, by building upon existing industry good practice in an efficient and effective manner"* to safeguard patient safety, product quality, and data integrity.

This *ISPE GAMP Good Practice Guide: Enabling Innovation* discusses critical thinking, incremental and iterative (Agile) software development models and methods, and IT service management. By applying *ISPE GAMP 5* [2] principles and current good practice, the GAMP CoP seeks to enable life-science industry innovation and advancement in these areas. This *ISPE GAMP Good Practice Guide: Enabling Innovation – Critical Thinking, Agile, IT Service Management* shows how these concepts are interwoven: applying critical thinking when leveraging software development practices, and using both concepts to effectively manage IT/IS service providers.

1.1 GAMP Guidance

This Guide is part of a family of guidance documents[1] that provide a comprehensive body of knowledge covering all aspects of GxP computerized systems good practice and compliance.

Figure 1.1: GAMP Documentation Structure



[1] For more information on the GAMP CoP and GAMP documentation, see www.ispe.org [3].

1.2 ISPE GAMP 5 Key Concepts

These five key concepts of *ISPE GAMP 5* [2] (also shown in Figure 1.2) form the basis of this Good Practice Guide:

1. Product and process understanding

   "An understanding of the supported process is fundamental to determining system requirements. Product and process understanding is the basis for making science- and risk-based decisions to ensure that the system is fit for its intended use."

2. Life cycle approach within a Quality Management System (QMS)

   "Adopting a complete computerized system life cycle entails defining activities in a systematic way from system conception to retirement."

3. Scalable life cycle activities

   "Life cycle activities should be scaled according to system impact on patient safety, product quality, and data integrity,…system complexity and novelty," and supplier capability.

4. Science-based Quality Risk Management (QRM)

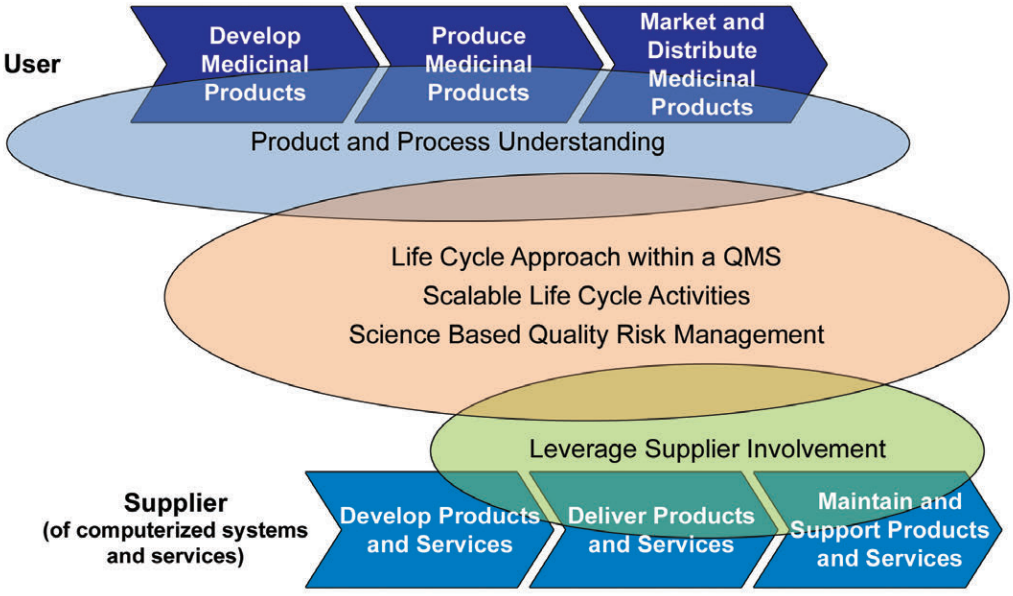
   "Quality Risk Management is a systematic process for the assessment, control, communication, and review of risks."

   The *ISPE GAMP 5* QRM approach is aligned with ICH Q9 [4].

5. Leveraging supplier involvement

   Regulated companies should seek to leverage supplier involvement and activities throughout the system life cycle while appropriately assessing and managing suppliers to minimize risks to patient safety, product quality, and data integrity.

Figure 1.2: ISPE GAMP 5 Guide Key Concepts [2]



1.3 Overview

This GAMP Good Practice Guide seeks to enable life-science industry innovation and advancement through the application of *ISPE GAMP 5* [2] principles and current good practice to the following key topic areas:

- The application of critical thinking
- Incremental and iterative (Agile) software development models and methods
- IT service management, including cloud computing

Chapter 2: Critical Thinking for Computerized Systems provides practical guidance on the adoption of a risk-based approach to the life cycle of GxP computerized systems based on critical thinking. It discusses how critical thinking may be applied to understand the intended use of the computerized system in the context of supporting GxP-business processes and the GxP data life cycle. Critical thinking supports the GxP principle of designing in quality (through system design, configuration, and operational controls) rather than the attempted testing in of quality.

Critical thinking allows the effective interpretation of data and situations while avoiding personal bias, inappropriate assumptions, and other distorting factors to retain rigor and balance.

As part of the Case for Quality program [5], the US FDA CDRH (Center for Devices and Radiological Health) has identified that an excessive focus on compliance rather than quality may divert resources and management attention toward meeting regulatory compliance requirements rather than adopting best quality practices. This may also deter investment in automation and digital technologies, which could assist in quality improvements and process control. An element of the FDA CDRH Case for Quality program is to promote a risk-based, product-quality and patient-centric approach to Computer Software Assurance (CSA). This encourages critical thinking based on product and process knowledge and QRM. [6] Chapter 2 discusses how critical thinking based on product and process knowledge and QRM can support such an approach. The application of critical thinking can enable the removal of barriers to the introduction of new and innovative technologies.

Chapter 3: Adopting Agile Software Development in a GxP Environment provides practical guidance on the adoption of iterative, incremental, or evolutionary (Agile) approaches to developing GxP computerized systems. Topics discussed include organizational culture, requirements management, specification and verification, and the use of effective tools and automation.

ISPE GAMP 5 [2] does not impose barriers to the adoption of iterative, incremental, and evolutionary approaches, and the specification and verficiation process described in it is not inherently linear. The FDA CDRH Case for Quality program [5] supports the adoption of Agile approaches where appropriate, in order to encourage innovation, eliminate unnecessary costs, and focus on quality and fitness for intended use.

Chapter 4: IT Service Management discusses good practices for assessing and managing IT/IS service providers. Regulated companies may utilize a diverse range of service models from IT/IS service providers to provide IT infrastructure, software, and data services. Delivery of these services range from traditional outsourcing models to Infrastructure, Platform, and Software “as a Service” (collectively referred to as XaaS or cloud computing). The appropriate implementation of cloud computing provides regulated companies with increased opportunities to adopt innovative methods to support the business. The application of these approaches requires careful consideration of the roles and responsibilities of the regulated company and the IT/IS service provider.

- **Accountability** entails ownership of the activity or task and being answerable if it is not completed correctly
- **Responsibility** entails doing what is necessary to ensure the activity or task will be completed correctly

Responsibility for IT-related activities may be delegated to IT/IS service providers, but in all cases regulatory accountability lies with the regulated company.

The regulated company must have defined roles and responsibilities for acceptance and release of GxP computerized systems. When outsourcing or delegating activities, there should be no resultant decrease in product quality, process control, or Quality Assurance (QA). There should be no increase in the overall risk of the GxP processes. The competence and reliability of service providers must be ensured.

Even though regulated companies cannot delegate their regulatory accountabilities to an IT/IS service provider, they may leverage the knowledge, experience, activities, and artifacts of an IT/IS service provider through risk-based assessment, management, and governance processes. A discussion of good practices in these areas is provided.

2 Critical Thinking for Computerized Systems

2.1 Introduction

Life cycle management of computerized systems in the pharmaceutical industry must ensure reliable operation and regulatory compliance while responding to the realities of maintaining and growing business profitability. While *ISPE GAMP 5* [2] presents a risk-based approach to ensuring fitness for intended use, some practitioners apply unthinking and rigid table-driven or tick-in-the box approaches not tailored or proportionate to the needs of different systems. Such suboptimal approaches often waste time and effort on non-value-added activities, leading to insufficient or excessive work with potential budget overspend and delays, and could inhibit innovation and the adoption of new technologies. Furthermore, these approaches may reduce focus on more valuable and essential QA activities.

The adoption of new and improved approaches to software engineering, data management, and "as a Service" offerings (XaaS), including the use of supporting tools, facilitate making the best use of resources and encourage the application of appropriate and proportionate practices. There have been schemes within the life sciences industries to improve processes that develop, manage, and maintain systems and software. Computer Software Assurance (CSA) within the US FDA CDRH Case for Quality program reinforces the importance of taking a risk-based approach that is primarily focused on impact to public health. [5]

This chapter discusses the concept of **critical thinking** applied to computerized systems. Critical thinking is defined in *ISPE GAMP® Guide: Records and Data Integrity* [7] as:

> *"a systematic, rational, and disciplined process of evaluating information from a variety of perspectives to yield a balanced and well-reasoned answer."*

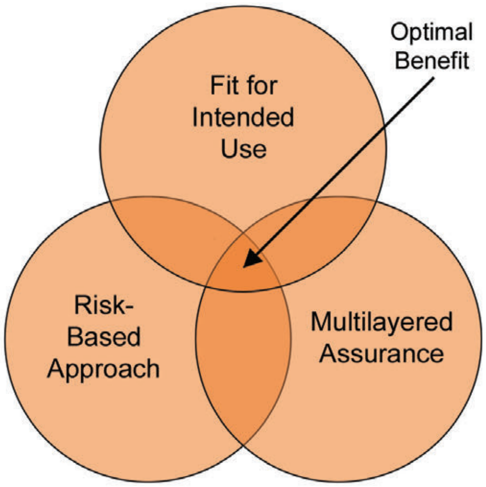
Critical thinking allows the effective interpretation of data and situations while avoiding personal bias, inappropriate assumptions, and other distorting factors to retain rigor and balance. Critical thinking is aligned with the application of ICH Q9 Quality Risk Management [4] principles. Critical thinking is not just a tool for individual decision-making. It can and should become a habitual mindset based on an intellectual commitment of using those skills to guide behavior. [8]

Figure 2.1 illustrates critical thinking for computerized systems. The critical thinking mindset promotes the proactive adoption of a risk-based approach suitable for the intended use of the computerized system that takes into account the multiple layers of assurance provided by the business process. In other words, combining technical, procedural, and behavioral controls applied throughout the business process when assessing the risk of the computerized system. These layers of assurance may exist upstream or downstream of the system within the business process it supports.

This holistic approach requires an initial investment of time and effort to analyze the overall business process that the computerized system will support, and the associated regulated data. Business process mapping and data flow diagrams capture this information and facilitate the identification and understanding of the potential risks to patient safety, product quality, and data integrity to determine where assurance is most needed.

Critical thinking supports informed decision-making and good judgment on where and how to scale quality and compliance activities. The extent and depth of activities (and level of documentation formality – see Section 2.3.6) can, and should, vary to a considerable degree between different business processes, types of systems, functions within a system, and applications. It is here that significant efficiency improvements can be made by focusing on what is really needed and avoiding unnecessary work.

Figure 2.1: Critical Thinking for Computerized Systems



Critical thinking can bring benefits throughout the life cycle of a computerized system. The effectiveness of initial planning for a computerized system is dependent on the quality and extent of critical thinking applied. If the intended use of a computerized system is better understood, requirements and specifications are better defined, then the rigor of testing different aspects of the system will match the importance of the system functionality. If a supplier has demonstrably tested a software or system function, what is the real value proposition of repeating this test? Similar benefits can be found in the operation and maintenance of systems and management of data. Critical thinking is not a one-time activity and should be evident during the computerized system life cycle all the way through to retirement.

Regulatory authorities are adopting critical thinking to help them more quickly determine whether controls are fit for intended use to ensure patient safety, product quality, and data integrity. Practitioners should not consider that the level of regulatory compliance achieved is directly proportionate to the amount of paperwork produced. Indeed, too much paperwork can confuse and make it harder to maintain and inspect computerized systems. Rather, regulators look for scaled and targeted activities with well-organized information and records that have an appropriate level of detail, supported by clear and unambiguous rationales explaining the critical thinking applied. The information/records contained within software development and support tools now offer the opportunity to demonstrate control in areas where separate documentation was previously considered a necessity, and these should be leveraged. [9] See Section 2.3.10 on inspection readiness.

Critical thinking involves questioning assumptions and eliminating bias. Bringing these into the risk-based approach can make the assessment process more objective. Critical thinking relies not just on basic knowledge of the business process but also on the detailed comprehension and analysis of where the business process can potentially impact patient safety, product quality, and data integrity. Critical thinking generates a better understanding of risks resulting in more confidence in the prioritization of those risks, and therefore supporting robust scaling of controls and validation activities based on that prioritization.

The combination of controls and validation activities provides multi-layered assurance. Critical thinking requires depth of understanding of the system within the context of the business process, such that the controls within the system complement the controls upstream and downstream in other systems to provide overall control. For example, the value of strong, validated data integrity technical controls in the system could be negated by errors caused by manual transcription of regulated data into the system. Data integrity cannot be recovered once lost.

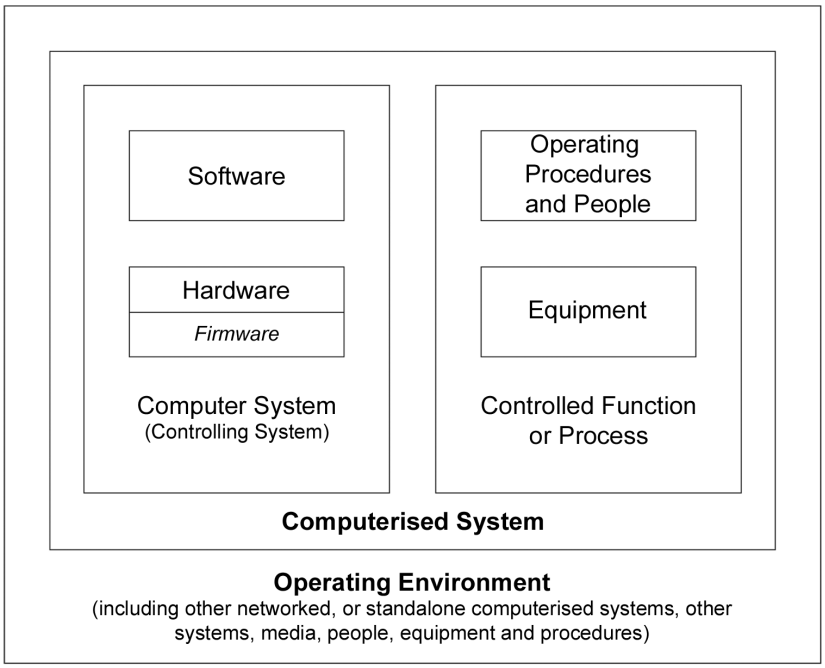
The benefits offered by critical thinking come from the experience and knowledge of the Subject Matter Experts (SMEs) specifying, testing, managing, and maintaining computerized systems. The SMEs can identify how best to realize the opportunities offered by critical thinking within a regulated company. The key objective is that computerized systems are fit for their intended use and efficiently maintained in a state of control.

2.2 Scope

A software application or computer system cannot be considered in isolation from its intended use, its role with the data life cycle of GxP data associated with the system, any connected ancillary equipment or interfaces, and its operating environment. This is defined in the PIC/S guidance Good Practices for Computerised Systems in Regulated "GxP" Environments [10] in 2007, as shown in Figure 2.2.

The definition of a computerized system is emphasized because a lack of precision in terminology may create confusion. Uses of the terms "application" or "computer" in place of computerized system does not limit the scope of computerized systems to software only. Software (i.e., the application) is only one component within a computerized system, which in turn is part of a wider operating environment. Failure to include all components of a computerized system in the scope of implementation, validation, and operational activities could result in not addressing potential risks to patient safety, product quality, or data integrity.

Figure 2.2: Components of a Computerized System [10]



2.2.1 Computerized Systems Life Cycle

A computerized systems life cycle approach means defining and performing activities in a systematic way from conception (understanding the requirements needed to support the intended use), through development, release, and operational use, to system retirement. The life cycle is scaled based on a risk-based approach with multiple layers of assurance to ensure the computerized system is fit for intended use. The project or development approach within the life cycle may be linear, iterative, or incremental.

Influences such as the increased adoption of Agile software development methods (see Chapter 3), Software as a Service (SaaS) application offerings and the associated software release cadence, and the need to update systems more frequently as part of cybersecurity efforts, mean there will increasingly be more changes during the operational phase of the life cycle. The effort needed to maintain the controlled, compliant, and validated state of the system through changes should not be a barrier to implementing upgrades so long as effective and efficient approaches are used.

Critical thinking combined with automated tools can facilitate efficient change management and regression testing. Automated test tools can verify critical functionality and ensure that regulated data is not adversely impacted after an application upgrade. Appropriate performance metrics can confirm solutions continue to be fit for intended use while safeguarding patient safety, product quality, and data integrity throughout the changes.

Chapter 4: IT Service Management discusses the advantages of moving from a fixed qualification approach for infrastructure to a continuous control and monitoring approach, which has the flexibility to accommodate and manage more frequent changes.

2.2.2 Data Life Cycle

A description of the data life cycle can be found in the *ISPE GAMP Guide: Records and Data Integrity* [7], and further discussed through three subsequent *ISPE GAMP RDI Good Practice Guides* [6, 11, 12].

Just as all components of a computerized system must be considered as part of fitness for intended use, so too must the aspects of the data life cycle that the computerized system supports. Each regulated record has an associated data life cycle where the data is created, processed, reviewed, reported and used, retained, and ultimately destroyed.

Note: Data and system life cycles operate on independent but related timelines. For example, a system may be retired before some of the most recent data within that system reaches the destruction phase of the data life cycle. This situation is discussed in Section 2.3.11.

Multiple systems may be involved in supporting a single data life cycle. For example, data may be created and processed in one system, reported and used to make GxP-decisions in an Enterprise Resource Planning (ERP) system, and then archived for the retention period in another. A data flow diagram derived from business process mapping illustrates all of these and helps to identify the regulated-data life cycle.

Areas to apply critical thinking to the data life cycle include, but are not limited to:

- Assessing the regulated-data life cycle holistically, considering all the systems and the interfaces between them that have the potential to impact the integrity of the GxP data
- Ensuring the system's user requirements reflect the technical controls required for the specific stages of the data life cycle that the system will support. For example, an archive system only supports the retention phase of the data life cycle, and therefore does not require controls for creating and processing data.
- Identifying what data needs to be archived and how to transfer that data if the archival solution is not the original system
- Understanding the diminishing value of data as it moves through the retention period and how that impacts the controls required
- Planning the means to view archived data if the original system that created the data is decommissioned

Mergers, acquisitions, and divestments can change data ownership and disrupt the data life cycle. Practical considerations for this are discussed in Section 3.7 of the *ISPE GAMP® RDI Good Practice Guide: Data Integrity by Design* [6].

2.3 Applying Critical Thinking

2.3.1 Risk Management

QRM, as defined in ICH Q9 [4] and adopted by *ISPE GAMP 5* [2], is:

> *"a systematic process for the assessment, control, communication, and review of risks"*

It is an iterative process used in the computerized system life cycle from concept to retirement. The objective of QRM is to identify risks to patient safety, product quality, and data integrity, and to apply appropriate controls to reduce those risks to an acceptable level.

Applying critical thinking to the risk-management process enables organizations to appropriately scale data and system life cycle activities and controls.

Figure 2.3: *ISPE GAMP 5* Quality Risk Management Process [2]

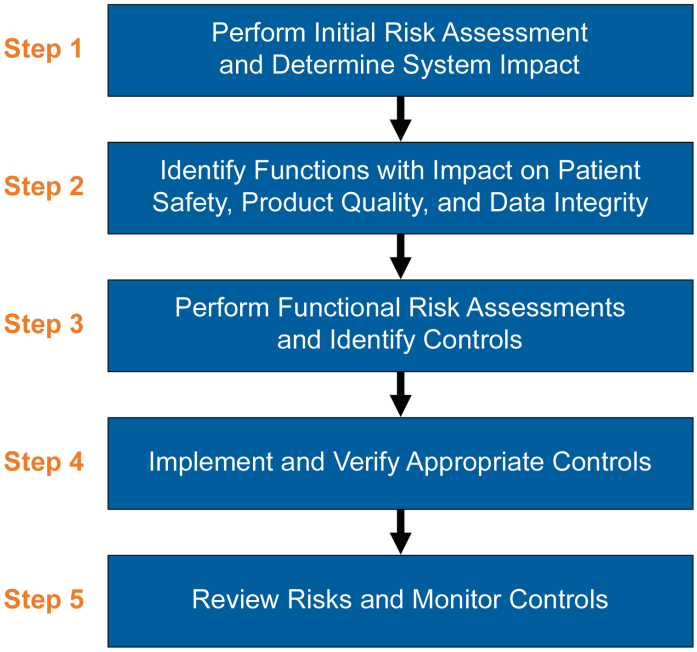


Figure 2.3 depicts the QRM Process described in *ISPE GAMP 5* [2]. Many organizations have focused on applying Step 3 of this process (the functional risk assessment that derives an overall risk priority based on severity, probability, and detectability) to all system functions without first adequately addressing Steps 1 and 2, and without applying critical thinking.

Step 1 provides the decision whether to proceed to Step 2 based on the overall impact that the computerized system may have on patient safety, product quality, and data integrity due to its role within the business processes.

Step 2 refines the scope of future activities based on which functions/features have an impact on patient safety, product quality, and data integrity. Functions not having an impact do not need to feed into Step 3.

Step 3 is the functional risk assessment that includes only those functions having an impact on patient safety, product quality, and data integrity. Functions identified in Step 2 should be assessed by considering possible hazards and how the potential harm arising from these hazards may be controlled. Understanding of the potential impact is aided by defining the business process and data flows, and assessing risk based on the overall business process, not just on an individual step in a system. Critical thinking should be applied to ensure the assessments are as effective as possible. For example, by assessing at a modular level, then at a business process level, and then at a functional level.

When assigning a rating for severity of harm to a potential failure, it is important to consider the overall risk of the system at the business process level. A complete failure to meet a requirement may render part of the system nonfunctional, but in a system with low overall risk to GxP, it probably does not rate as a high severity of harm with respect to patient safety, product quality, and data integrity. There could, however, be justification for classifying severity of harm as high with respect to its impact on the organization's other operating imperatives.

Similar requirements relating to a specific area of functionality may be grouped together in a recursive hierarchy consisting of major requirements and subsidiary requirements. This hierarchical approach, which may have multiple levels depending on the complexity of the process or system, may help simplify requirements management and risk assessment. Major requirements[2] are assessed based on severity, probability, and detectability to derive an overall risk priority (High, Medium, or Low per *ISPE GAMP 5* Appendix M3 Figure M3.5 [2]).

Critical thinking and the major requirement risk priority (e.g., is it high risk?) should determine whether some or all of the more detailed subsidiary requirements need to be individually assessed within the hierarchy of functionality. A subsidiary requirement cannot have a risk priority higher than that of the major requirement, so there may be no benefit to assessing subsidiary requirements of a low-risk priority major requirement. However, for a higher-risk major requirement, identifying subsidiary requirements with lower-risk priority prevents the need to automatically test all subsidiary requirements. Combining this approach with a testing strategy based on increasing test rigor and documentation with increasing risk priority ultimately results in efficiency gains.

Assessing the risk at the major and/or subsidiary level as needed ensures the selection of optimal control measures (configuration settings, procedural controls, data review processes, etc.), and testing rigor and documentation, commensurate with the assigned risk priority.

Risk assessment allows the identification of appropriate behavioral, procedural, or technical controls to reduce risk to an acceptable level. They may be part of a computerized system function, or in parallel manual procedures, and may be upstream or downstream of the system. Controls typically are aimed at:

- Eliminating risk through process or system redesign
- Reducing risk by reducing the probability of a failure occurring
- Reducing risk by increasing the in-process detectability of a failure
- Reducing risk by establishing downstream checks or error traps (e.g., fail-safe, or controlled fail state)

The risk priority may be used as input to decisions on rigor and extent of controls. For example, application timeout and password complexity are both controls to reduce the risk of unauthorized access to a system. The duration of the timeout and the extent of password complexity may be influenced by the risk priority, with shorter timeouts and greater complexity where the risk has been assessed as high.

Step 4 implements and verifies the controls selected in Step 3 to ensure that they have been successfully implemented. Controls should be traceable to the relevant identified risks. The verification activity should demonstrate that the controls are effective in performing the required risk reduction.

[2] See Section 2.3.3 for examples of major and subsidiary requirements.

Some companies have a policy at the organizational level that provides guidance on the test strategies applicable to the different controls and risk priorities. Critical thinking can assess the supplier testing completed on the identified controls and determine what, if any, additional assurance is needed and what assurance approach(es) (or combination of approaches, e.g., ad hoc, exploratory, robust scripted etc.) is most appropriate based on the risk priority assigned in Step 3. Determining the appropriate assurance approach is discussed in detail in Section 2.3.7.

Step 5 is performing system periodic review or planning changes to the system. Critical thinking should be applied to evaluate trends in performance metrics, changes in the assessed risks, the impact of changes in the system, the need for changes to the implemented controls, and the need for new or regression testing activities.

It is not necessary to apply the steps in this risk-management process for nonregulated systems, although there could other business reasons to do so, such as if the system is critical to the ongoing operation and the organization wants to benefit from the risk-based approach in all areas.

The level of effort, formality, and documentation of the quality risk-management process should be commensurate with the level of risk. The effort involved in applying the risk-based approach should not be more burdensome than if it were not applied. Key stakeholders for risk assessments include IT, quality, the process owner, and SMEs skilled in computerized system validation, risk management, and critical thinking.

2.3.2 Planning

In order for an organization to implement a computerized system that supports their business process and ensures data integrity, they should use a holistic approach. Planning for individual systems is discussed in *ISPE GAMP 5* Section 4.2.1 [2].

A business process is often supported by multiple computerized systems. An organization may aim to improve the efficiency, data integrity, or regulatory compliance of the business process by looking to update one of the systems or consolidate multiple systems into a single enterprise system. Critical thinking must be applied throughout the process to understand the risks to patient safety, product quality, and data integrity, and in particular, for an understanding of the effect and risks of the implementation of the new computerized system.

To facilitate effective system planning, it is important to understand the business process and the data flow, including all regulatory requirements. Developing business process maps and data flow diagrams to illustrate business activities is described in *ISPE GAMP Guide: Records and Data Integrity* [7] and the *ISPE GAMP RDI Good Practice Guide: Data Integrity by Design* [6]. A full understanding of the regulated process to be supported, including the intended use of data within the process, is fundamental.

The business process flowcharts illustrate the business activities, decision points, and subprocesses, while the data flow diagrams identify the creation, movement, use, and archiving of data. Together these diagrams help in the understanding of the business process to identify risks to patient safety, product quality, and data integrity associated with the process. Data integrity cannot be achieved without a complete understanding of the data flow.

Examples of planning using critical thinking include:

- Utilizing business process mapping and data flow diagrams to understand where the computerized system will fit in the process, what regulated data will pass through it, and what part of the regulated data's life cycle the system will support
- Defining consistent nomenclature for use in the process to facilitate data transfer, trending, and analytics
- Selecting a solution that best fits the business requirements minimizing configuration and customization, recognizing that it may be more pragmatic to adjust the business process to fit a standard application

- Understanding the interfaces needed to other systems, and what standard interfaces are available versus developing new interfaces (with their additional test and management burden)

- Planning for how the system should behave during normal operation as well as how it should respond in an error or failure situation and how it can recover from such a situation

2.3.3 Requirements

The key to the implementation of a computerized system fit for intended use is to thoroughly understand the business process and data requirements to enable the creation of user requirements. Not all requirements need to be finalized before proceeding (for further information on Agile approaches see Chapter 3). Requirements should define the functionality required to support the business process as well as ensure compliance with applicable regulations, for example, data integrity. Guidance on defining data integrity requirements is contained in *ISPE GAMP Guide: Records and Data Integrity* [7] and *ISPE GAMP® RDI Good Practice Guide: Data Integrity – Key Concepts* [11].

Critical thinking should ensure that requirements specifically relating to regulatory compliance are tailored to the system's intended use rather than indiscriminately applying every potentially applicable regulatory reference when some are not appropriate or necessary. For example, following the text of the regulation for an audit trail requirement can result in oversized tables of text entries that are dependent upon manual review. Applying critical thinking initially evaluates whether a data audit trail is appropriate or necessary, in other words, are users expected to create, modify, or delete regulated records during normal operation? If an audit trail is required, critical thinking should be used to develop requirements (including Agile user stories) for an audit trail that is contextual, searchable, filterable, and reportable. In this way, critical thinking enables the development of more efficient and effective controls.

Increasing levels of requirements granularity can be expressed and managed by using a recursive hierarchy within the specification. In the example above, the major requirement would be:

- The system must have a time-stamped, computer-generated audit trail

The subsidiary requirements then describe how to ensure effective and efficient controls. Some simple examples to illustrate this hierarchical approach are:

- The audit trail will record operator entries and actions that create, modify, or delete regulated data (e.g., within the batch record).

- The audit trail shall be searchable by any of the recorded information, e.g., user ID, description, date, time.

- An exception report can be generated from the audit trail, checking for user-defined criteria.

The subsidiary requirements may not necessarily be specifically defined in the regulations but may significantly impact the usability of the system. As noted in Section 2.3.1, critical thinking may warrant the risk assessment of individual subsidiary requirements of a high risk major requirement to better scale the controls and the testing effort.

Developing defined requirements is an essential precursor to identifying those with the potential to impact patient safety, product quality, and data integrity, and to applying QRM approaches, as discussed in Section 2.3.1.

2.3.4 Supplier Assessment and Selection

By performing assessment, management, and governance activities, regulated companies should ensure that suppliers, including service providers, will meet the regulated company's technical, business process, and regulatory requirements. Confidence that a supplier offering will support the business process needs to be based on assessing their capability for developing, implementing, and maintaining a system fit for the regulated company's intended use. In the case of "as a Service" offerings, this should include assurance of the long-term provision and stability of that offering. The need for supplier audits (initial and ongoing) should be based on risk. Supplier assessments should determine whether the supplier is suitable for use.

Knowledge of the processes followed by the supplier can enable the regulated company to reduce their validation effort by leveraging supplier activities, as described in *ISPE GAMP 5* [2] and *ISPE GAMP RDI Good Practice Guide: Data Integrity – Key Concepts* [11].

Critical thinking combined with a clear understanding of the business process enables evaluation of the computerized system's functionality and the supplier's specifications. Rather than just noting the presence of a development life cycle during a supplier assessment, applying critical thinking during the assessment establishes the transparency and traceability of the supplier's internal development life cycle to determine how much of the supplier activities can be leveraged. This may require participation from the regulated company's business process owner, compliance, and technical experts during the assessment. Further discussion of how critical thinking can be applied to reduce user validation of Commercial off-the-Shelf (COTS) products has been published elsewhere [13].

It is important to understand the origin of the supplier's software modules and libraries. It is becoming common to request a *Software Bill of Materials* to understand the various components and any use of Open-Source Software (OSS).

Dependence on, and trust in, a supplier is not based on a single assessment but rather aggregated throughout the system life cycle and includes reassessments. Critical thinking should ensure that supplier selection criteria contain performance measures such as system reliability, service continuity, and reputation for customer responsiveness. For certain types of system architecture, such as SaaS, reliance on the supplier may not be limited to the system life cycle and can include the data life cycle, which should be assessed during service provider selection.

2.3.5 Impact of Supplier on Regulated Companies

Maximizing efficiency between suppliers and regulated companies requires critical thinking from both parties, even when the supplier is in-house to the regulated company.

Where a supplier is creating an offering to the life sciences industry (whether that be a control system element such as a Programmable Logic Controller (PLC), analytical laboratory equipment, IT software solutions, or a SaaS application), it will have its own life cycle where it is planned, specified, implemented, verified, and reported as fit for intended use as part of an internal release process. A supplier's offering may be used by a regulated company *as is*, parameterized, configured, or customized to meet the requirements of the business process.

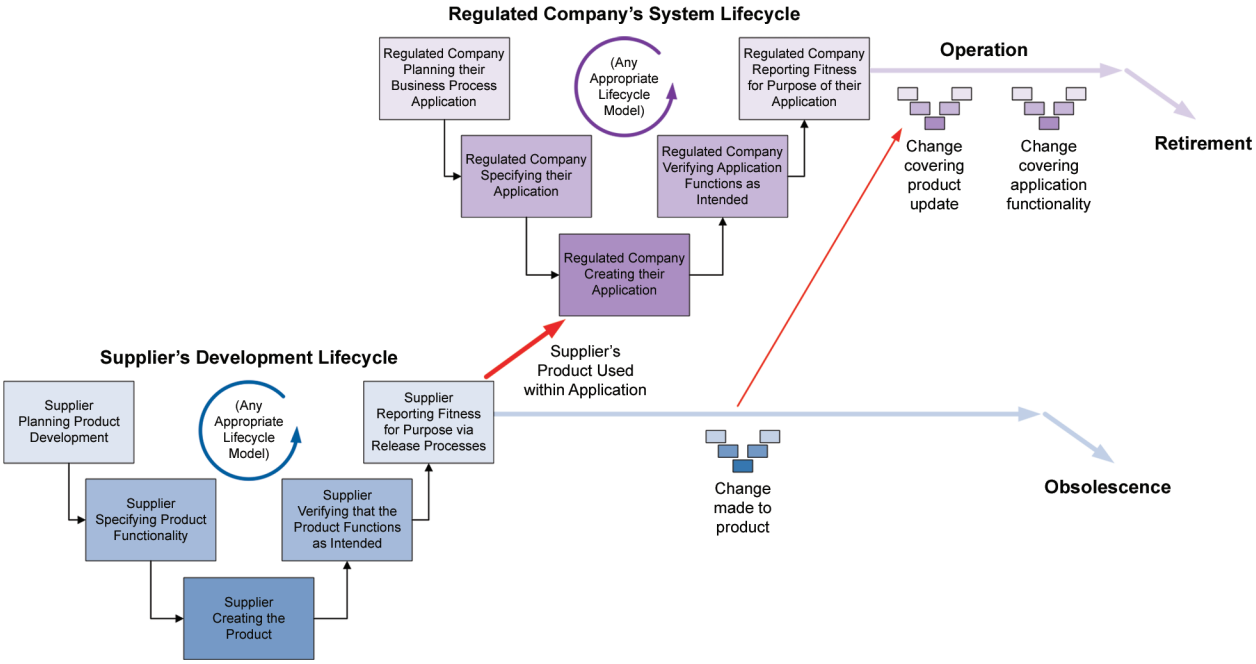
The supplier should apply a risk-based approach to their development life cycle, including their test strategy. Supplier activities should focus on ensuring their product meets their specifications by effective and efficient means; however, there will be a minimum level of detailed information and evidence of effective testing expected by a regulated company. For example, critical thinking would identify that ineffective testing approaches instead of robust and/or automated testing may reduce the opportunity for the regulated company to leverage the supplier's activities. Where the supplier has used a risk-based approach, the regulated company should determine if their assessment of risks is different to that of the supplier and adjust accordingly.

The better the quality of the supplier's development processes (e.g., supplier's **development** life cycle), the more efficient the regulated company's implementation of the supplier's offering can be (regulated company's **system** life cycle), as depicted in Figure 2.4.

Neither of these life cycles are a single event. The supplier will release updates to their products and services, and the regulated company will need to assess the updates and, if implementing them, use operational change control to apply the updates. Where a regulated company has in some way customized the offering, the effort to implement the update may be proportionally larger.

In an "as a Service" offering, particularly multitenant SaaS applications, the use of a DevOps approach[3] can result in more frequent changes with little or no opportunity for the regulated company to assess or reject a change. It is inherent to the service model that the development life cycle may drive the validation schedule and change control. Managing and leveraging service providers is discussed in more detail in Chapter 4.

Figure 2.4: Interaction of Supplier and Regulated Company Life Cycles
Adapted from ISPE GAMP® Good Practice Guide: A Risk-Based Approach to Testing of GxP Systems (Second Edition) [14]



Critical thinking should be applied to ensure that:

- The supplier's understanding of how their product is used in the life sciences industry allows them to specify and test their product in a way commensurate with its potential risk to patient safety, product quality, and data integrity.
- Regulated companies are accountable for ensuring the system is fit for intended use, and should evaluate the supplier's approach against the regulated company's intended use. Access to supporting information around supplier activities should be defined in the purchase contract or Service Level Agreement (SLA).
- Regulated companies should leverage the supplier's information describing the functionality of the product so that they can cross reference to their own requirements. This enables the regulated company to analyze the functionality and identify any areas that need configuration or customization to suit their business process. They should also seek to leverage the activities performed by the supplier.
- Information and test evidence in the form of artifacts within requirements management tools and automated test tools are given equal credence with discrete formal documentation. Further considerations for the use of tools are discussed in Section 2.4.4.1.
- If the supplier's system is to be interfaced to other systems within the regulated company, for example LDAP (Lightweight Directory Access Protocol) authentication or single sign-on, the interfaces need to be defined, implemented, validated, managed, and maintained.

[3] A DevOps approach combines software development and system operations and is often used to support continuous delivery.

- Where feasible, the supplier should aim to develop customer-requested new functionality into their mainstream product offering rather than creating special versions for individual customers. This reduces the supplier's product management effort and simplifies future updates for all parties. If the functionality is provided as a configurable option, there is no obligation on other customers to implement it.

- Where a regulated company requests the development of a new or amended feature, keeping it within the supplier's development life cycle ensures it is done once and tested once rather than developed and tested repeatedly by regulated companies.

- Reuse of standard modules of code or custom-developed forms, by the supplier or regulated company, should be based on a master template to ensure changes are automatically replicated across all instances. This reduces the possibilities of errors (e.g., introduced by copying and pasting for reuse), condenses the need for testing to just the master template, and eliminates the possibility of one or more instances not being updated.

- Ongoing management and implementation of changes is done in a controlled way by:

  - Leveraging supplier activities and knowledge (i.e., in the same way as that for the original development).

  - Ensuring changes to the product can be communicated to the regulated company so they can assess the risks of taking or not taking the change. (Note that in some offerings the change is not optional for the regulated company, e.g., multitenant SaaS).

  - Having a pre-verified method for installing the change. The regulated company should apply critical thinking around the risks of taking versus not taking the change, assessing likely risks to their intended use as a result of the change, and leveraging what has been done by the supplier in order to install the changes with the minimum of repeated effort.

- Where a supplier is involved within the regulated company's life cycle (e.g., a systems integrator), the supplier and regulated company should jointly agree on the overall rigor of documentation and testing required and who will be responsible for which elements in order to avoid unnecessary duplication.

2.3.6 Plans, Specifications, and Reports

The specification and verification approach described in *ISPE GAMP 5* [2] for project phase activities has been interpreted as a linear-sequential approach. In fact, the model is not confined to linear-sequential approaches and has much greater value as a relationship diagram. Irrespective of the software development process followed and the validation strategies adopted, the key objectives of the project phase of the system life cycle are that the verification activities are traceable to the system specifications, and the reporting stage identifies compliance or noncompliance to the system requirements and validation plans.

ISPE GAMP 5 [2] and *ISPE GAMP Good Practice Guide: A Risk-Based Approach to Testing of GxP Systems (Second Edition)* [14] provide detailed guidance on testing strategies and documentation. *ISPE GAMP 5* states that the supplier's specifications and test evidence should be assessed for suitability, accuracy, and completeness. It specifically states that *"there should be flexibility regarding acceptable format, structure, and documentation practices"* as there is no value in recreating documentation in a different template.

Critical thinking can be used to determine:

- Scaling activities and records based on risk

- If the supplier's specifications and evidence are adequate for leveraging to support the regulated company's system life cycle, and available when needed

- What information needs to be directly available at the regulated company versus referencing the supplier's records

- Who should be the vital few reviewers/approvers and in what context are they reviewing/approving. In practice, having more approvals usually leads to lower quality and value as everyone thinks someone else has looked at the detail. Segregation of duties must be preserved where required.

The application of risk-management approaches determines the areas of highest risk within the computerized system. Critical thinking then enables tailoring the level of formality for test plans and specifications for the highest risk areas. For example, not all testing records have to be to the same level of detail. The use of critical thinking for testing is covered in Section 2.3.7, and the use of tools is discussed in Section 2.4.4.

2.3.7 Testing

A key principle from testing theory is that exhaustive testing is impossible [15], and consequently testing can only show the presence of defects and cannot prove the absence of defects. Early and efficient testing, during development as well as during the verification phase, improves defect detection and reduces the occurrence of defects surviving into the operational phase.

Risk-based approaches focus test effort on high-risk functions and inherently do not test everything; therefore, there is the possibility that some defects will not be detected. Regulated companies should apply critical thinking when investigating escaped defects discovered in the operational phase. An excessive response may be to initiate a test all approach to find other escaped defects. Instead, test metrics should be used to confirm that the actual escape rate of defects is in line with the assumptions made when determining the type of testing to be carried out for functions and features with different risk priorities. An unacceptable number of escaped defects in high-risk functions requires re-evaluating the functional risks and the corresponding test strategies.

Another concept from testing theory is the Pesticide Paradox[4] [16] whereby based on the results of testing, programmers improve their programming practices (i.e., fix the defects) so as to make previous test cases ineffectual at defect detection.

In order to avoid test specification deviations, testers often dry run protocols to ensure they will run without test script errors during the formal execution because the consequences of a test scripting/authoring error are disproportionately high. The test cases are run until they have little possibility of detecting a defect, providing no value from a testing perspective. This focus on error-free test scripts takes time away from testing without benefit to patient safety, product quality, or data integrity.

Regulated companies should think critically on strategies to minimize the bureaucracy associated with correcting largely inconsequential script errors so that the dry run activity can be minimized. Errors that impact product quality will trigger detailed investigations compared to a *catch and correct in the moment* approach for less significant typographical errors. Testing techniques such as unscripted and ad hoc testing (discussed in Section 2.4.4.2) can dramatically reduce the incidence of script errors, although depending on risk priority, a level of scripted testing may still be required. Unscripted testers should be qualified as testers and trained, competent users of the system to ensure the testing is focused on challenging the application in the context of the business process. Test activities should be traceable to the requirements.

2.3.7.1 Approach

ISPE GAMP Guidance [17] promotes a risk-based approach to GxP computerized systems compliance and fitness for intended use. The FDA CDRH Case for Quality program [5] reinforces the need to take a risk-based approach to testing and to focus testing effort on functions or features that present risk to patient safety, product quality, and data integrity while using critical thinking to ensure efficient test approaches.

[4] The Pesticide Paradox – If the same tests are repeated over and over again, eventually these tests no longer find any new defects. To detect new defects, existing tests and test data may need changing, and new tests may need to be written. Tests are no longer effective at finding defects, just as pesticides are no longer effective at killing insects after a while. In some cases, such as automated regression testing, the pesticide paradox has a beneficial outcome, which is the relatively low number of regression defects. [16]

A combination of direct evidence (e.g., manual or automatic recording of results, screenshots) and indirect evidence (e.g., reports, notifications that can only exist if the proving step is completed) can be used to establish that requirements have been fulfilled in support of the intended use.

By applying critical thinking throughout the life cycle (see the other subsections within Section 2.3), the functions and features to be tested can be defined with appropriate risk priorities applied that reflect:

- The potential impact on patient safety, product quality, and data integrity (high priority functions with the greatest risk require greater test rigor and level of documentation detail)
- Prior testing of the function or feature either in the supplier's development life cycle or as part of the regulated company life cycle. Automated tools within the supplier's development life cycle may have enabled and included full regression testing of daily builds.
- The degree of confidence in that prior testing based on supplier assessment

Using critical thinking, those risk priorities can then be used as input to determine the appropriate test approaches. The aim of testing is to identify and allow the removal of defects and confirm fitness for intended use rather than producing documentation for documentation's sake. Critical thinking can optimize test approaches for the regulated company such as:

- Planning and organizing tests to run as efficiently as possible, for example by combining tests to minimize repeated test setup activities, and/or by grouping related functionality testing
- Ensuring sufficient test coverage during the system life cycle, with the rigor of testing commensurate with the assigned risk priority, and avoiding repeating tests that are similar or identical to those carried out by others
- Ensuring that the test cases demonstrate that functions operate correctly
- Differentiating between proving steps and non-proving steps to limit the amount of test execution evidence created and retained. Proving steps demonstrate a requirement has been fulfilled, while non-proving steps are used to set up the proving step. This ensures test evidence is only captured for proving steps that demonstrate a higher-risk GxP or business requirement has been fulfilled.
- Making test cases specific and unambiguous such that a reviewer can objectively determine if the test passed or not without subjective analysis or excessive evidence (see Section 2.3.7.2)
- Ensuring the testers have the skills and expertise to execute the tests, operate the system, and evaluate the results in the context of the intended use and/or the business process. This then obviates the need for overly prescriptive and detailed test instructions and in turn reduces test incidents arising from poorly written test scripts.
- Minimizing pressure on testers, e.g., to meet a deadline for moving to the operational phase, which can bias testing outcomes
- Enabling a proportionate review of completed tests based on the risk priority of the function under test
- Leveraging automated test tools and test management tools in place of extensive manual effort and referring to the test artifacts within the tools in place of documentation (see Section 2.3.7.2 for examples of reducing test effort by using automated testing, and of reducing test results documentation by leveraging the audit trail entries generated during automated testing)
- Managing incidents and their corrective actions to ensure any changes are fully verified once completed, including an appropriate amount of regression testing to ensure that the change has not adversely affected other functionality

Test strategies should ensure sufficient system testing to detect defects, and that test management processes are robust enough to maintain control of the system during the testing activities. Some possible strategies for scaling test rigor and documentation in the regulated company's system life cycle are discussed in greater detail in *ISPE GAMP RDI Good Practice Guide: Data Integrity by Design* Appendix S2 – Computer Software Assurance [6].

The critical thinking rationale behind the test strategy must be documented.

2.3.7.2 Test Evidence

Testing activities and test evidence should focus on demonstrating fitness for intended use by showing that quality-critical requirements have been met, identifying defects so they may be removed, and ensuring that risk-management controls are effective.

A combination of test approaches (such as exploratory, error guessing, manual/automated, unscripted and scripted testing) is needed at various stages in the system life cycle in order to achieve these objectives. The test evidence or output varies from case to case and depends on the detailed testing technique and intended readership.

The organizational policy, or the test specification itself, should define the requirements for test evidence, including the recording and reporting of test pass/fail results. *ISPE GAMP 5* Section 8.5.3 [2] identifies excessive and unnecessary hard copy test evidence as an area for efficiency improvement. In particular, it acknowledges the significant overhead associated with hard copy evidence and reiterates the need for a:

> *"justified and documented decision, based on impact, novelty, and complexity"* of the system.

In many cases, additional hard copy evidence such as screen shots do not add value and are unnecessary.

The objective evidence for the testing performed may include the application data and objects created during testing and can leverage the system audit trail, if it has been verified to accurately capture input, transactions, and outcomes in sufficient detail. The audit trail also enables the company to generate and retain further detailed test evidence when using unscripted testing techniques. Supplemental evidence (e.g., reports, screenshots) can be captured when it adds value beyond the audit trail, for example, investigating errors, facilitating the ease of review, documenting the complexity of the transaction.

Test evidence is inherent in the application data and audit trail entries created during testing (documented unscripted testing is discussed in Section 2.4.4.2), which can be reviewed as required. Additionally, the system's previously verified functionality can be demonstrated as fit for intended use within the test environment or production environment as applicable.

Critical thinking around applying a risk-based approach to the extent of test evidence supports:

- Ensuring that even where testing is not scripted in detail, the objective of the test remains clear, and the results record adequately what testing was carried out. (Unscripted does not mean undocumented – see Section 2.4.4.2.) Evidence of a test case result can be as straightforward as the tester indicating if a test passed or failed.

- Ensuring that test evidence is only collected for proving steps that are not inherently covered by evidence from another test.

- Adopting an exception-reporting approach to recording detailed results. That is, that if the system response matches the expected results, a simple "Pass" can be recorded. However, if the system responds in an unexpected manner, the tester records both a "Fail" and a description of how the response differed from the expected results, as this additional detail is helpful in determining the root cause and corrective action. This eliminates time wasted capturing elaborate test evidence (e.g., excessive screenshots or recording detailed descriptions of the observed response) when a simple pass or fail (or recording the value or initialing a design specification statement) is adequate to confirm that the test has been completed.

- Recognizing that screenshots as objective evidence may now be considered as bringing little value to verification activities. Note that there are a number of situations explained in the *ISPE Good Practice Guide: A Risk-Based Approach to Testing of GxP Systems (Second Edition)* [14] where they offer a practical benefit over manual recording when detail is needed, e.g., when there is a specific need for a before and after comparison of detailed or complex data, such as an audit trail or report.

- Acknowledging that it is disproportionate and unnecessary to use test witnessing or requiring the tester to initial every test step to affirm they followed the instructions

- Encouraging the use of automated testing tools to minimize manual collection of evidence as the artifacts within the tools are themselves evidence

2.3.7.3 Reviewing Completed Testing

Completed testing should be reviewed to verify that the computerized system is fit for its intended use. Test evidence should be reviewed to confirm completeness and accuracy of conclusions.

There is little value in:

- Looking for minor errors in documentation that have no impact on patient safety, product quality, or data integrity

- Forensically scrutinizing test evidence to assess whether the tester has exactly followed the test instructions for non-critical functionality

Unusual patterns of test failures associated with particular authors and/or individual testers should be investigated to determine whether there are any wider testing implications on the rigor of completed testing.

2.3.8 Operation and Maintenance

The operational compliance of a computerized system – and hence the safeguarding of patient safety, product quality, and data integrity by that system – is significantly impacted by the working practices of its users. Critical thinking and user consultation across both the business users and the IT support group Standard Operating Procedures (SOPs) should be applied to develop a framework of easily understood and intuitive practical instructions (e.g., SOPs, video how-to guides, built-in online help) for routine use and data review, system support, incident management, and system administration, etc.

The configuration choices and controls implemented during the project phase will only be realistically exercised during the use of the system. Such controls should be reviewed, often in periodic review (discussed in Section 2.3.9), using critical thinking to assess whether they are truly mitigating the risks as intended, and whether they still enable efficient business operation. Changes to system configuration are typically captured into a system audit trail that should be regularly reviewed to ensure no unauthorized changes have been implemented.

Detailed information about how to maintain computerized systems throughout their operational phase is discussed within *ISPE GAMP 5* [2] and *ISPE GAMP® Good Practice Guide: A Risk-Based Approach to Operation of GxP Computerized Systems* [18].

Since those guides were published, changes in technologies and development methods have led to an increase in release cadence for applications. The regulatory expectation is that application updates that offer improvements to patient safety, product quality, or data integrity should be applied when available and that GxP systems do not run on unsupported operating systems. [9]

This requires regulated companies to implement operational controls as a flexible and streamlined process, leveraging critical thinking and risk-based approaches, to manage and maintain the system and its compliant use through ongoing changes and evolution during its operational life. This allows change to be embraced in support of improved operation rather than avoided as long as possible because of the perceived validation and documentation burden associated with change.

Increasing incidences of cyberattacks have forced the industry to routinely adopt patches and hotfixes immediately after release to address potential security vulnerabilities. Managing IT infrastructure changes proficiently is addressed in Chapter 4.

When a system has been tested using a risk-based approach, it is inherent that a residual level of risk has been accepted including the possibility that there may be defects, particularly in low-risk functionality, that were found during testing and are still present in the system. When a defect is discovered during operational use and the incident management process triggered, it is important to apply critical thinking during the root cause analysis to assess whether the original risk-based testing approach is still valid. This is discussed in Section 2.3.7.

2.3.9 Periodic Review

During periodic review the system is assessed for the cumulative impact of any changes, defects, or regulatory updates. The frequency of periodic review should be based on the GxP impact of the system, with high-risk systems reviewed more frequently. Critical thinking should be used to determine whether the review frequency should be increased or decreased based on the outcome of the previous review and/or performance trends. For example, if issues and failures were found during the last periodic review, the review period can be shortened to make sure the Corrective and Preventive Actions (CAPAs) were completed and have resolved those issues.

Additionally, periodic reviews may be triggered by other processes such as performance monitoring, incident management, and CAPA. Periodic reviews will be needed more frequently if the system is not managed using performance monitoring metrics. Critical thinking should also be used in conjunction with system performance metrics and trending data to identify areas for improvement within the system's use, maintenance, and ongoing operation. This includes considering whether incremental changes to the system's functionality has extended the system beyond its intended use.

Critical thinking can be applied to ensure that the appropriate rigor is applied to change management to minimize the frequency of periodic reviews and ensure the validated state is maintained, thus eliminating the need for revalidation.

Periodic review is referred to in Section 2.3.1 and is more extensively discussed in the *ISPE GAMP Good Practice Guide: A Risk-Based Approach to Operation GxP Computerized Systems* [18].

2.3.10 Inspection Readiness

Having applied critical thinking, it is vital that this thinking and the subsequent approach taken to various aspects of the life cycle of a computerized system are evident for regulatory inspection. Policies and procedures can position where critical thinking will be applied, and plans and specifications prompt specific considerations during its application. The practical application of critical thinking, however, also needs to be evident in the rigor of the system activities. For example, rationales should be available to explain how high risks were identified. These rationales can then be used to justify the level of documentation between different aspects of functionality and the corresponding amount of evidence collected for testing and qualification. It is worth remembering that a regulator will view the computerized system with a fresh pair of eyes, and obvious viewpoints and decisions may not be self-evident several years after the matter when someone not involved in the initial activity looks at it.

During an inspection, a regulated company may need to provide evidence of the assessment and qualification of their supplier [19]. Where supplier activities and information are leveraged as part of the regulated company's validation for intended use, consideration should be given to the arrangements needed that ensure information on supplier assessment and management processes is available for review.

Holding duplicate copies of supplier information that rightly belongs in the supplier QMS is unnecessary and brings the risk of inconsistency and complexity. The regulated company may consider contractual agreements allowing access to critical supplier documentation under specified extreme circumstances.

For clinical trial specific systems, the EMA Q&A: Good clinical practice (GCP) [20] states that the sponsor or Clinical Research Organization (CRO) must have

> *"detailed knowledge about the qualification documentation and can navigate in it and explain the activities as if they had performed the activities themselves."*

There must also be established configuration management procedures and documented justification of differences between the supplier's validation environment and the sponsor's production environment.

2.3.11 Retirement

Retirement is the last of the system life cycle phases and consists of withdrawal, decommissioning, and disposal. In some situations, a computerized system may be withdrawn from active operations but not immediately decommissioned or disposed of, if it is needed to read retained records/information. This should not be treated as a long-term solution to record readability because systems and software become obsolete over time.

The *ISPE GAMP Good Practice Guide: A Risk-Based Approach to Operation of GxP Computerized Systems* [18] details the steps and considerations to complete the retirement, decommissioning, and disposal process.

A key element with planning system retirement is consideration of the data associated with that system. Critical thinking and risk management are needed to effectively evaluate what data needs to be retained, for what period of time, and how, such as:

- Data originally created in this system that has completed its retention period and, as long as there are no legal holds associated with it, may be deleted from the system and/or archived via a formal process.
- Data that is inactive at this stage, i.e., only likely to be needed in an investigation or audit. Based on risk, it may be acceptable to retain this data in a static, easily readable, and portable format.
- Data recently created in the system and may need to stay dynamic for a period of time. Critical thinking is needed here to determine the most efficient way to achieve this and may involve balancing the risk of data migration against the complexity of maintaining a legacy copy of the system.

Data migration may be chosen as a solution to data readability after system retirement to get the existing data into the new system. Data migration is discussed in some detail in *ISPE GAMP 5* [2] and *ISPE GAMP Good Practice Guides* [17]. Automated migration tools provided as standard items by system suppliers are often either assessed in great detail or are used without considering the types of problem data[5] that might occur within the end-user's system. The regulated company needs to consider the maturity of the supplier, the robustness of the tool, and how well the tool meets the migration requirement. The assurance of the tool should be focused on actual use cases including any identifiable problem data needed to confirm that it migrates correctly.

Automated test tools may be also leveraged to test the migrated data. Critical thinking is needed for planning how to cleanse and verify that the data is ready for migration, how to assess the quality of the migrated data, how much data needs to be checked, and how any errors created during the migration process may be detected.

[5] Problem data here is meant data with missing content, data in an incorrect format, data that cannot be handled by the target application, etc.

2.4 Practical Considerations

2.4.1 Organizational Capability

People are the source of critical thinking. Critical thinking will only work when used by skilled SMEs with sufficient experience and knowledge.

Organizations need to consider how to build and grow their critical thinking skills within the business. This includes specifically seeking critical thinking skills when recruiting new personnel, incorporating critical thinking into training and development, and ensuring there is a critical thinking element within the balance of team formation.

The mindset and culture of an organization also needs evolve to support critical thinking. Open and constructive discussion between stakeholders is vital to evaluate and challenge the situation, the information, assumptions, and organizational precedents in order to make better decisions based on agreed rationales.

The capability of an organization needs to be proactively grown to become more efficient and effective in its application of critical thinking. The capability build can be measured and consequently steered through a series of levels of increasing maturity, for example:

- Level 1: No application of critical thinking evident in decision-making either by practitioners or by management
- Level 2: Some awareness of critical thinking within the organization but highly variable across individuals and departments
- Level 3: Critical thinking is described in policies and procedures but is inconsistently applied
- Level 4: Critical thinking principles are fully incorporated and routinely applied in working practices
- Level 5: Critical thinking is respected as a core competency with organizational capability continually improved

2.4.2 People, Process, and Technology

Guidance around computerized systems tends to focus on technical controls (i.e., software features and functionality) that can be validated as performing as intended. Once validated, the regulated company can be confident that the control will operate to prevent or support the relevant activity, as applicable. However, there is more to a computerized system than software (see Section 2.2.1).

Critical thinking should consider the computerized system within the context of the business processes it supports and the combination of people/process/technology involved. Compliance and fitness for intended use in a process or system requires addressing all components of a computerized system including, but not limited to, its operating environment and interfaces to other systems, the process it supports, and the people operating the process.

The process component of a computerized system can cover (but is not limited to):

- Statistical analysis of clinical trial data
- Process Analytical Technology (PAT) and other manufacturing monitoring and control processes
- Complex analytical laboratory assays
- ERP applications
- Simple workflow with procedural controls

Some of these require validation in their own right – analytical method validation, process validation, etc. – to adequately safeguard patient safety and product quality.

There is also always a level of human interaction present in a process: the people impact. This could be as minimal as simply initiating the process or could include the option for manual intervention or override in the process, which substantially increases risk. Critical thinking should evaluate the level of possible human intervention and its potential impact on the integrity of the data. People controls (procedural and behavioral) are discussed in Section 2.4.3.

2.4.3 Data Governance and Controls

The application of critical thinking skills help ensure data governance and supporting processes are both effective and proportionate. A holistic approach should be taken that includes the technical, procedural, and behavioral controls needed to achieve and maintain data integrity.

Technical controls for data creation, audit trails, access control, data transfers, and storage and archiving are key to effective data management, and are detailed in *ISPE GAMP® Guide: Records and Data Integrity* [7]. Critical thinking for technical controls should consider the controls in other systems or services that support data integrity throughout the data life cycle. With increasing use of external service providers, ownership, retrieval, retention, and security of data needs to be understood. Quality agreements established with third parties need to define the objectives that must be met without being prescriptive as to the methods. An ongoing relationship needs to be established, applying critical thinking and information feedback between the parties so that the objectives are met overall across the business process. Procedural controls should exist as part of the QMS that include policies, procedures, and supporting templates and guidance. Personnel should have the training, facilities, and time to fulfill their roles.

Data integrity is highly dependent upon personal behaviors. Critical thinking plays a key role in establishing effective behavioral controls, recognizing that strength of quality culture varies across different locations driven in large part by geographic values and local historical context. A single approach to quality management and data governance may not be effective in all situations, for example, if cultural differences and dynamics challenge the acceptability of openly reporting problems and challenging hierarchy. Codes of conduct should specifically state behavioral expectations to ensure the reliability and completeness of data, explaining the benefits this brings to patients and the personal consequences for employees who breach controls.

The holistic perspective prompted by critical thinking should ensure the work environment uses technical controls to facilitate simple and effective ways of working. Critical thinking also helps ensure supervisory measures are proportionate to risk, such as, where is it appropriate to have a contemporaneous second person verification versus a later check of completed data activity. Special consideration is needed for hybrid situations where records are compiled from manual and automatic processes to ensure controls are complete and complementary without being overly bureaucratic.

Critical thinking can be used to help identify and target appropriate levels of training for different audiences. Training should cover:

- Human factors for design and use of computerized systems
- Reporting data integrity issues
- Transparent investigation of incidents and issues to understand true root causes beyond the initial reasons so that robust computerized systems are created and maintained

The content of training also needs to be scaled to the detail and reinforcement of principles needed. Training is not necessarily effective if it just repeats earlier content already delivered to the same audience. Trainers should consider fresh and innovative approaches to make training interesting and informative so that its effectiveness is maintained and enhances organizational capability.

The *ISPE GAMP Guide: Records and Data Integrity* [7] and supporting *ISPE GAMP RDI Good Practice Guides* [6, 11, 12] provide more detailed guidance on the practical application of critical thinking to data management.

2.4.4 Tools and Techniques

2.4.4.1 Requirements Management and Traceability Tools

Requirements management and traceability tools can make development, implementation, and maintenance of systems more efficient and less error prone. Such tools provide a collaborative workspace that defines requirements, develops test cases in tandem with requirements coding, and tracks issues and defects. Relational databases within these tools can be leveraged to manage the traceability of the requirements to validation activities throughout the life cycle. Such tools can make traceability more efficient and proactive.

Where sufficient level of detail and approvals are contained and available within the tool, then there is no benefit to patient safety, product quality, and data integrity for manually creating separate documentation as audit evidence. Reports extracted from the tool in common portable formats (e.g., requirements, test outcomes, and traceability matrices) can be generated on request.

Requirements management tools, traceability tools and automated test tools should have a documented assessment of their adequacy (see also Section 3.5). Good IT practice should ensure that the data/records held are controlled, secure, and available.

2.4.4.2 Unscripted and Scripted Testing

Unscripted testing is testing in which, while the testing activity is still documented, the tester's actions are not prescribed by detailed instructions in advance of test execution to the same extent as scripted testing.

Typical approaches include ad hoc testing, error guessing, and exploratory testing. Details of the testing performed, by whom, and outcomes and conclusions are still recorded in all cases.

- **Ad hoc testing** is unscripted testing performed without planning or pre-defined documentation. It is aimed at finding defects as early as possible.

- **Error guessing** is a testing technique designed to expose anticipated and potential defects based on the specialist tester's knowledge and experience of failure modes.

- **Exploratory testing** is experience-based testing where the tester spontaneously designs and executes tests based on existing specialist tester's knowledge and experience, prior exploration of test item (including results from previous tests), and typical common software behaviors and types of failure and defects.

Unscripted testing, therefore, does not mean undocumented testing. While such tests do not have step-by-step instructions, they still record what was tested, by whom, when, and any issues found. Unscripted testing, when used as part of the overall test strategy, can have significant advantages over traditional scripted testing. By not being overly prescriptive in how a test objective should be met, testers will attempt different paths to prove that the functionality is working correctly and is robust when it comes to fault handling. This naturally tests more than a one-path approach.

Unscripted and scripted testing are both valuable tools in the same way a hammer is neither better or worse than a screwdriver: each has its purpose and should be deployed appropriately. Scripted testing provides the affirmation that the system is fit for intended use, whereas unscripted testing is more focused on defect detection. The test strategy should use critical thinking to combine the different types of unscripted and scripted testing to complement each other for maximum effectiveness.

A comparison of unscripted and scripted testing approaches is provided in Table 2.1, which is an evolution of earlier CSA work [21].

Table 2.1: Comparison of Unscripted and Scripted Testing Approaches

Unscripted Testing	Scripted Testing
Used to supplement scripted testing to improve defect detection on high-risk functions. Critical thinking could support using only unscripted testing to challenge functions with low and/or medium risk.	Used alone or in combination with unscripted testing approaches to rigorously challenge functions with high risk to patient safety, product quality, and data integrity.
Used to uncover software defects or errors associated with poorly defined/implemented specifications.	Tests against specifications to confirm fitness for intended use. Scripted testing, manual or automated, can provide the basis for regression testing to capture the impact of changes or updates.
Relies on the testers intuition, knowledge, and testing experience to explore and challenge the functionality of an application.	Test cases use the specification as the standard to be verified. By its very nature scripted testing may not uncover defects arising from poorly defined specifications as scripted testing follows the specification verbatim.
Aims to test both expected and unexpected user/system behaviors. Use of an inexperienced tester and/or a lack of system knowledge negates many of the benefits of unscripted testing.	Tests are designed to confirm expected user/system behaviors.

Unscripted techniques such as exploratory testing allow the behavior of the system to determine the path forward. Exploratory testing leverages the Plan – Do – Check – Act cycle.

> *"Testers simultaneously learn about the product and its defects, plan the testing work to be done, design, and execute the tests, and report the results. Good exploratory tests are planned, interactive, and creative."* [22]

As the tester learns the application, they are better able to explore and test the application without preconceived notions about its behavior that come from scripted test cases. Unscripted testers should be qualified as testers and trained on the system in question to ensure the testing is focused on challenging the application in the context of the business process.

Another form of unscripted testing, leveraging the intended system users, is to adopt a "day in the life" approach. This testing practice allows users of the system to carry out their normal day to day activities in the system (as they would post go live) to uncover issues not identified by scripted testing. These users may not necessarily have a software testing background but instead have the real-life business process experience and knowledge, and can work through the system in conjunction with its routine use SOP to identify defects that impact the normal operation of the system. Such testing also identifies work arounds or poorly designed user interfaces. This can be important to ensure system controls cannot be easily bypassed.

Given the reduced documentation for ad hoc and unscripted testing, test planning needs to ensure the combined coverage of scripted and unscripted testing addresses the requirements. Additionally, an exploratory testing charter could be generated prospectively, and/or the completed test records or logs could be reviewed retrospectively to confirm adequate requirements coverage was achieved and is traceable. Such traceability could be recorded in the test plan or in a traceability matrix. The level of rigor around the nature of the documentation and approvals is determined by the organizational culture. Appendix S2 Computer Software Assurance in the *ISPE GAMP RDI Good Practice Guide: Data Integrity by Design* [6] contains additional guidance on the use of unscripted testing for GxP computerized systems.

2.4.4.3 Managing Build and Configuration

The traditional "static snapshot" approach of installation qualification and configuration management has proven difficult to apply when using modern virtual environments and cloud computing. If the build is maintained in a controlled environment with regular checks to certify the system remains within its specified setup condition, then the need to independently confirm the installation is reduced to a review by exception. Most automated build installers provide reporting tools for any configuration and setup anomalies and failures with a dashboard to help operational staff quickly respond and resolve any issues found. Such configuration management of the build script ensures that the continuous certification is synchronized to compare the current system against the correct setup conditions. Critical thinking can be applied to justify using modern processes such as this, since technology has superseded the need for traditional aspects of qualification. Chapter 4 covering IT service management discusses this in more detail.

3 Adopting Agile Software Development in a GxP Environment

3.1 Introduction

Agile software development approaches focus on delivering quality and value to the customer at speed, and in an incremental fashion, thus enabling technical innovation and flexibility.

Agile software practices have been established for decades, and are widely adopted by many industries, including medical devices. The use of Agile within other areas of the pharmaceutical industry has been more limited, even though the majority of software development for the base applications and major configurable products used to support GxP processes applies Agile approaches. This chapter provides a brief overview of the principles underpinning Agile and illustrates how it can be implemented in a way that is completely aligned with GxP and *ISPE GAMP 5* [2] principles.

The key principle behind Agile software development is that of **discovery** and of **iteration** (ongoing changes). This differs from a waterfall approach. With waterfall software development, there is a **linear flow** of defining/collecting all requirements before transforming them into a complete set of functional and design specifications that are then coded/configured before testing commences. With Agile software development, requirements are collected and then moved into development/configuration, testing, and release in iterative cycles. This typically requires the ongoing involvement of cross-functional teams including end users and business process owners.

There are a number of differing frameworks that support Agile, however all of these are underpinned by the principles of the Manifesto for Agile Software Development [23]:

> ***"Individuals and interactions*** *over processes and tools*
>
> ***Working software*** *over comprehensive documentation*
>
> ***Customer collaboration*** *over contract negotiation*
>
> ***Responding to change*** *over following a plan"*

These principles need to be taken in the context of "emphasis" rather than a binary choice. For example, tools are invariably used, and form an important part of Agile software development, but the manifesto stresses that team collaboration is more important than simply tools/processes.

Scrum is one example of an Agile framework, and can be summarized as:

- A Product owner is responsible for collecting requirements into a product backlog. Section 3.4 covers requirements gathering in more detail, however the key elements defined are typically epics[6], which are larger sets of requirements and are further split into sets of (user) stories.
- The Scrum cross-functional team takes a (prioritized) subset of the backlog items and develops/delivers and tests these during a sprint (a short, time-constrained period to complete a set amount of work). This is referenced against a defined Definition of Done (DoD) – a list of requirements that a user story must meet to call it complete.
- A review is conducted of the results of the sprint (the sprint retrospective),which is used to drive continuous improvement in the working method/process.

[6] Epics and stories are explained in Section 3.4.2.

- The cycle then repeats with another subset of the backlog.

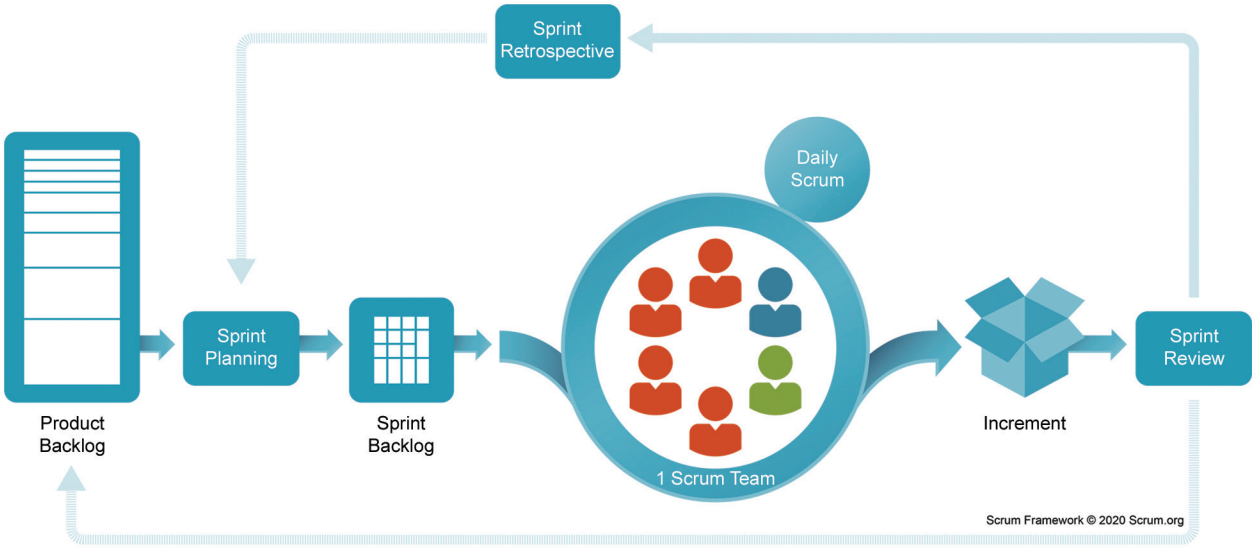
Sprints, also referred to as development sprints, are governed by time and designed to be a short period of time for the Scrum team to deliver. For example, 2 to 4 weeks could be a reasonable period for a sprint and any backlog items unable to be completed during the sprint return to the backlog.

Scrum teams typically are small, and for Scrum, only three roles are defined: Scrum Team, Scrum Master, and Scrum Product Owner. The team needs to be multidisciplinary and is responsible as a whole for delivering to time and quality. The scrum product owner provides the customer link (e.g., to the business process owner) and the Scrum master provides coordination and helps ensure the team adheres to the rules/governance/processes agreed to by the team. Scrum team members need to have expertise across roles such as developer, tester, architect, and quality but people are encouraged to work across boundaries in order to enhance their skills and knowledge.

Regulated company quality roles typically provide oversight and (in line with the critical thinking and risk-based approach) subject matter expertise on regulations and potential areas of product quality, patient safety, and regulated-data impact associated with the business process the system will be supporting.

The Scrum process is illustrated in Figure 3.1.

Figure 3.1: Scrum Framework Model [24]
Used with permission from Scrum.org, www.scrum.org.



In combination with the above model are a DoD and Minimum Viable Product (MVP). These are defined in order to determine the items/activities to be completed before a requirement (epic or user story) can be considered complete and to agree/define the minimum set of requirements (epics/user stories) that need to be complete for the initial release of the product into use. Agile is typically used in combination with sets of tools that provide the necessary controls over the product backlog, configuration, testing, and release activities. Due to the iterative nature of Agile software development, there is a risk of newly developed functionality impacting the correct operation of previously developed and tested software, and therefore regression testing (often performed using automated test tools) is typically applied to ensure stability of the system within sprints.

The specification and verification approach described in *ISPE GAMP 5* [2] is not inherently linear, and the approach described is designed to be compatible with a wide range of other models, methods, and schemes including incremental, iterative, and exploratory models and methods (see Figure 3.2).

Figure 3.2: ISPE GAMP 5 Specification and Verification Showing Iterations

ISPE GAMP 5 [2] describes the overall GxP system life cycle from the perspective of the regulated company and does not define the software development process in detail.

ISPE GAMP 5 supports the use of incremental, iterative, and evolutionary approaches, including Agile, for product development and development of custom applications. [25] Factors for successful adoption include a robust QMS within an appropriate organizational culture, well-trained and highly disciplined teams following a well-defined process supported by effective tools and automation, and proper customer or product owner involvement.

Some potential misconceptions around Agile and GxP, and consequences and inefficiencies that can arise from these are:

- Suppliers using Agile for product development and then being directed to provide deliverables that resemble the waterfall approach to the regulated company
- The Agile Manifesto [23] statements taken too literally. For example, "working software over comprehensive documentation" does not mean no documentation or records.
- References to documentation interpreted within a very narrow context of traditional approved documents/specifications rather than seeing these also as records/information/artifacts within software tools.
- Interpretation of the *ISPE GAMP 5* [2] high-level project stages of Plan, Specify, Verify, and Report/Release as linear/waterfall only and not realizing that iteration between stages is natural and acceptable.
- Agile seen as not in control due to poor or lack of adherence to the method or ceremonies.[7] When correctly executed, Agile is a controlled process, and just as in the waterfall approach, if executed poorly and with a lack of controls, is unacceptable.

[7] *"Agile ceremonies are periodic meetings held to ensure that projects are on time and meeting quality goals."* [26]

In some cases, good Agile software development activities are performed with the necessary deliverables (often these are records rather than documents), which are then reverse engineered to produce a suite of traditional documents. This is wasteful and potentially confusing as two sets of deliverables need to be maintained/aligned through the system life cycle. It also raises the risk of negative product quality or patient safety impact if misalignment between documents and tools prevents information in documents from getting into the toolset.

One aspect to carefully consider is the scenario where a supplier and regulated company are working together on the development of a system. Given the typical dependency of Agile methods on tools, and the minimization of documentation in favor of the information/records and reports from such tools, where there is to be a handover of responsibility (e.g., subsequent support and configuration) from the supplier to the regulated company, it is important that the handover includes how any tools will also be included. Ideally this should be considered early in the contractual discussions.

It may be tempting to try and map artifacts created using Agile methods to the traditional validation documents (for example mapping user stories/epics to the User Requirements Specification (URS)); however, it is more effective to start from established good practice Agile artifacts and not force-fit the traditional documents. Mapping against the high-level project stages as illustrated in Figure 3.2 is sufficient, and many activities, such as a supplier assessment, user training, and periodic review, are conducted irrespective of the development method used.

The content of this chapter is therefore aligned to a risk-based approach, applying critical thinking per the principles described in Chapter 2 and avoiding solely bureaucratic approaches/activities and deliverables that do not add value.

The FDA CDRH Case for Quality program supports the adoption of appropriate Agile approaches in order to encourage innovation, eliminate unnecessary costs, and help focus on quality and fitness for intended use. [25]

3.2 Scope

Agile software practices are typically applied to bespoke/custom systems (i.e., GAMP Category 5 [2]) where a new software application is developed to satisfy business process requirements, or during the development of standard or configurable commercial software products (i.e., GAMP Category 3 [2]). Agile concepts are, however, equally applicable for configured systems (i.e., GAMP Category 4 [2]) or very large systems where initial system deployment with limited functionality (but still providing business value) can be delivered earlier through the use of Agile.

Section 2.3.5 discusses the interaction of supplier and regulated company life cycles (see Figure 2.5). It is important to note that it is completely acceptable for these life cycles to be both Agile or a mixture of Agile and linear.

The decision as to which approach to adopt, in particular for the regulated company, is likely to be scenario dependent. For example, if the regulated company has a very clear set of initial (user) requirements and wishes to use these as part of the tender process, and to select the supplier and their product, then a linear approach for the regulated company may be appropriate. This does not preclude the supplier from using Agile, however, and user requirements in this scenario could be mapped against supplier user stories/epics and other Agile artifiacts to establish coverage and traceability.

It also depends on how knowledgeable the developers are about the proposed development and the technical aspects and risks. If the development is straightforward, and the developers have developed a very similar product before (e.g., the requirements, technology, and people are the same), then a traditional linear approach may be appropriate.

Where the regulated company is looking at less clearly defined scope/requirements, and especially for customized developments (including potentially in-house developments rather than with external suppliers), then Agile is likely to be the preferred approach to enable faster initial system deployment and subsequent incremental development.

In the first scenario a traditional URS would come from the regulated company, but tools and Agile approaches may still be fully applied by the supplier during technical activities, with the scope of the activities varying based on whether the system is predominantly standard, configurable, or custom.

This chapter provides an overview of four key concepts for consideration by organizations looking to adopt Agile in a GxP environment:

- The Discovery Mindset
- From Requirements to Product
- Tools Instead of Documents
- DevOps, Continuous Integration/Deployment and Product Teams

Other Agile considerations discussed in this chapter include the need to invest in tool sets and the scenario where third-party supplier development is to be handed over to the regulated company for ongoing maintenance/support.

3.3 The Discovery Mindset

Advancement in technologies and the speed of these changes, together with the uncertainty that these introduce, have forced organizations to rethink business models. Organizations that were once highly constrained, orderly, and slower in changing, with typically fixed processes for executing established activities, are now more dynamic and complex, serving patients and customers that are better informed and with higher expectations than ever before.

Now more than ever, organizations need to be patient/customer orientated and data driven. Patients have choices and are better informed. They have embraced a digital world with a huge market of wearable health technologies. They can go online and research symptoms, possible causes, and potential treatments and expect their apps and devices to constantly update with minor/micro changes. Patients expect intuitively designed products that do not need training courses to know how to use them, and they are quick to replace products that do not match these expectations.

To succeed in this dynamic, not only do work practices and tools need to change, but also people's behaviors and management systems, in order for the improvements to be sustained.

The discovery mindset is a powerful approach to changing the way teams plan and deliver. This also applies to business process owners and quality groups, where in a traditional approach, they tend to be involved at a detailed level at key stages (e.g., approve the entire URS, review all acceptance test results), whereas with Agile it is more a continual cycle of involvement against more manageable items of delivery through sprints.

On its own, the discovery mindset helps teams with continuous learning, but its real power comes from changing how success is measured and the mechanics of delivery. This can be seen, for example, by looking at the value delivered to the company and meeting the needs of patients and customers rather than the cost of delivery, or of frequent production releases instead of annual releases.

3.3.1 Mindset

"Mindset describes how the organization thinks. It's about how culture, values, and priorities manifest themselves throughout the organization." [27]

A **discovery mindset** encourages acting, learning, and quick improvement.

Waterfall or linear programs and projects have often been focused on the company's needs rather than the end users. Average release cycles were measured in quarters or years, generally being such large releases that users needed extensive and time-consuming training on the new systems.

There is a perception that people can estimate and predict the distant future, where end dates are fixed and often for large waterfall programs, the final testing phases are squashed and reduced because earlier phases over ran.

It is not linear approaches that are the issue, as there are many examples of successful linear projects. The issue is the **certainty mindset** and how it is tempting to think the world is certain, and not undergoing constant change, and it is this uncertainty that Agile handles so well. [27]

With the discovery mindset it is possible to embrace the constantly changing environment and the uncertainty of everything. The 2020 COVID-19 pandemic significantly changed and challenged every individual and every company across the globe. [27]

In order to achieve this discovery mindset, long-term goals need to be considered, but progress toward them should be through small effective changes based upon predicting and estimating the near future.

3.3.2 Culture and Principles

Every successful Agile adoption or transformation is specific to an organization's culture, history, technology, and people. Changes in work practices and tools, as well as people's behaviors and management systems, are needed in order for improvements to be achievable and sustainable. [27]

"However, there is a common set of principles that apply:

- *An agile-first mindset and ways of working, using a Discovery Mindset, rather than a Certainty Mindset*
- *A culture of innovation and continuous improvement*
- *Empowered teams with the ability to deliver across geographies*
- *A consistent and repeatable way of agile delivery*
- *Much-improved alignment across all stakeholders"* [27]

3.4 From Requirements to Product

Many regulated companies expect to see a fully formed URS when developing a GxP regulated system. This section aims to demonstrate that by planning and storing requirements differently and following Agile principles, teams can successfully deliver applications in a controlled way that is compliant with GxP regulations. What is required in Agile is that a complete and consistent set of requirements is defined (e.g., epics, user stories) and verified before the system is released for use in the GxP environment, which is a natural and achievable outcome of a well-controlled Agile process.

Successful delivery requires ensuring "We build the right system" that is fit for the intended use, not just ensuring "We build the system right." Traditionally, the project URS is interpreted as a must-have document containing the full set of requirements before any other work is done, regardless of what project approach is used.

This is the certainty mindset in action, as discussed in Section 3.3.1. However, EU Annex 11 [19] is aligned with the ISPE GAMP 5 Appendix D1 [2] statement that corresponds to the discovery mindset:

> *"Requirements may not initially be fully defined, e.g., for some Category 5 systems, requirements are developed during subsequent phases of the project."*

Requirements are identified in Agile within user stories, epics, or features, and managed on a backlog basis. Ultimately the delivered description of system functionality is provided by the list of completed Agile artifacts (such as epics and user stories), which can be obtained directly from, or as reports from, the Agile software development tools.

In Agile, having a full set of requirements at the start is not possible nor desirable. If there is a full set of requirements, the team would not be applying an iterative or exploratory approach, and probably not an incremental approach either.

An advantage of Agile is the use of tools, which provide the ability to add and change requirements, as well as the ability to include audit trailing, approvals, status, and traceability. Caution needs to be employed when considering the whole product backlog itself as a form of user requirements, as this can contain items such as defects and other types of work. Techniques such as defining the MVP and DoD or similar can be used to provide controls to ensure that initial and subsequent product releases satisfy the product owner and can be moved into production.

3.4.1 Where do Requirements Come From?

All companies have a demand pipeline of requirements, and while they may not explicitly talk about having this demand pipeline, it is there. New demand can come from many places, such as those helping to improve the business, and from strategic goals, initiatives, or value streams. Another source is run the business that provides feedback on the currently live system such as defects or minor enhancements. Functional and nonfunctional requirements may also be required to fulfill regulatory and quality requirements such as data integrity (for example audit trail requirements) and IT security aspects.

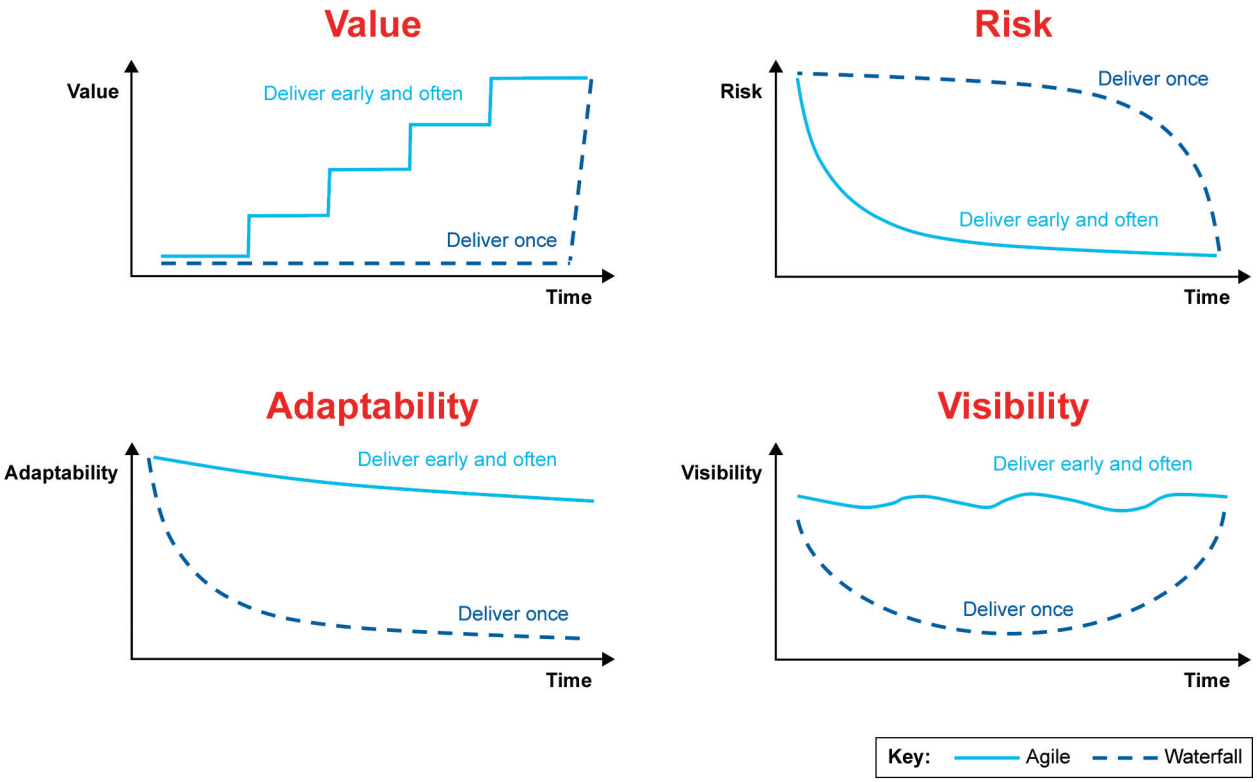
With Agile, teams embrace the discovery mindset to get early and frequent releases into production, repeatedly looping through the demand pipeline. Each release gives another chance to obtain feedback from real usage, to inspect and adapt, altering the plan and adjusting the system as needed. It allows teams to experiment with hypotheses to see if the resulting system is meeting the intended needs of the user, customer, or patient.

Figure 3.3 illustrates some advantages of Agile compared to the traditional waterfall delivery. Using Agile:

- Value is delivered earlier
- Risk is reduced and visible at a much earlier time
- System adaptability is facilitated

- Changes or new requirements do not cause the negative impact typically seen at later stages of waterfall/linear software development
- Users get early visibility and experience of a system

Figure 3.3: Agile Versus Waterfall Delivery Comparison Illustrations [28]
Used with permission from Emergn, www.emergn.com.

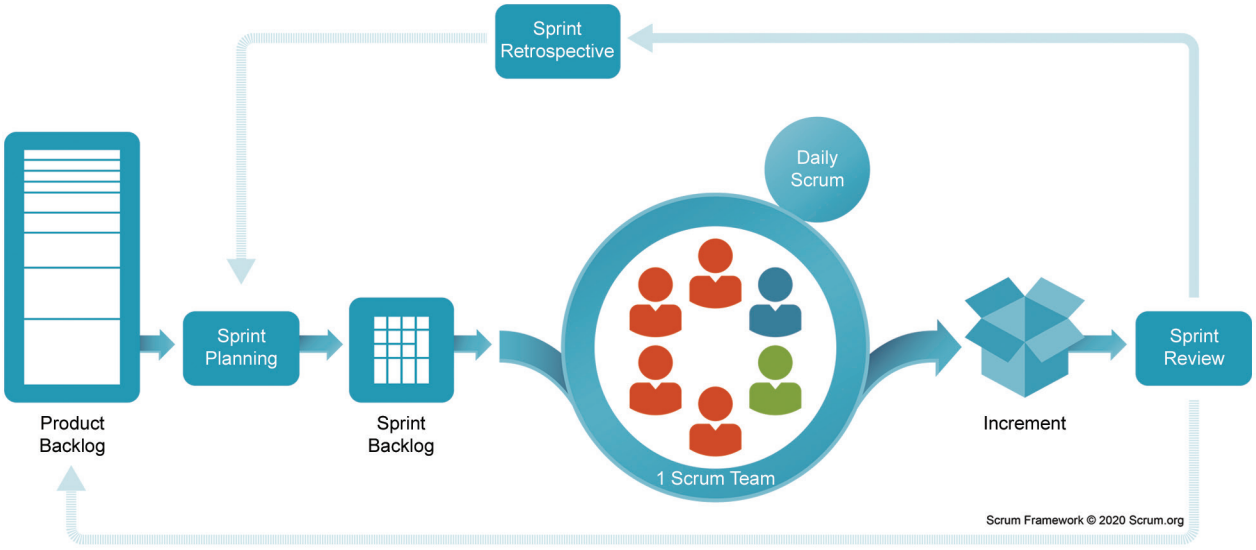


Early and recurrent releases can help deliver value sooner, giving feedback on how the early releases are performing, and with smaller releases inherently lowering the risks (compliance and delivery) associated with the usual large releases. They also help increase the team's adaptability and responsiveness to changing needs, all the while increasing visibility to stakeholders or customers as to how the team is progressing and providing earlier return on investment.

3.4.2 How Does Agile Handle Requirements?

Agile is a generic term consisting of a guiding manifesto and twelve principles. [23] This section discusses how the Scrum Framework documents requirements, how most Scrum teams differentiate requirements, and how they can be used within the GxP world. See Figure 3.4 for a process overview.

Figure 3.4: Scrum Framework Model [24]
Used with permission from Scrum.org, www.scrum.com.



Scrum takes a specific approach to requirements definition and management. Traditional URS take much time to produce, review, and approve, and subsequent changes or additions have to progress through the same typically slow review/approval processes. Instead, Scrum has the concept of a product backlog, usually stored in an Agile planning tool, providing benefits of digitalization such as filtering, reporting, audit trailing, and traceability.

The **product backlog** is a list of individual items that have been taken through a demand pipeline usually by the product owner. These **Product Backlog Items** (PBIs) can consist of various types of work, such as defects, epics, stories (sometimes called user stories), enhancements, etc., reflecting the product owner's current, and accepted, view of scope and priority of what needs to be done.

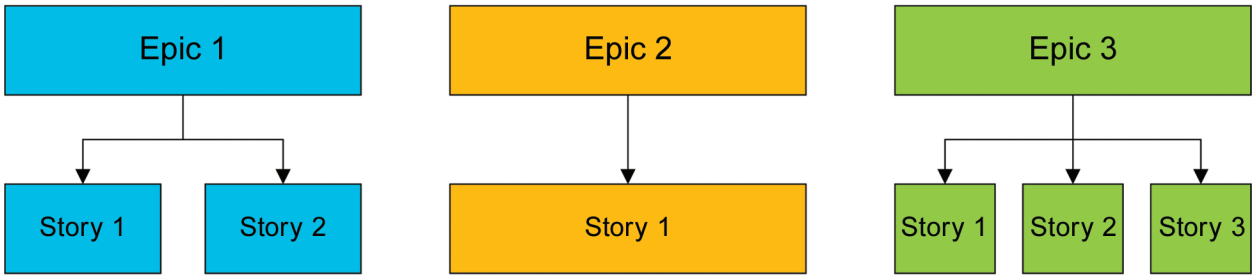
Most Scrum teams organize the product backlog by using epics as the largest item. Epics are then broken down into a set of small stories. Just as in the literary world, epics are large stories. In this case, large stories (or any other PBI) are those that cannot comfortably be delivered within a sprint.

All PBIs should have **user acceptance criteria** specified at a high level, in order for the development team to understand the intent of the PBIs and use them as a basis for potential test cases while working on the story.

A **sprint backlog** describes the list of tasks planned for the sprint to meet the purpose of the sprint, called the sprint goal. It is essentially a subset of the most important items from the product backlog.

For teams new to Scrum, one way to think of epics and stories is that an **epic** is like a user requirement (the what), and **stories** are its functional and nonfunctional requirements (the how), albeit, with a very different style of wording. Figure 3.5 provides an example of the relationship. Note that depending on the Agile framework and scale of the development, additional artifacts and terminology may be introduced/used, for example, initiatives, features, and tasks.

Figure 3.5: Epic-Story Relationship



An Example Epic:

- As a line manager, I want to control who can access our systems so that we can ensure only authorized and authenticated individuals are permitted.

Example Stories for the Epic:

- As a line manager, I want to provision new joiners (employees) so that they have a standard account.
- As a line manager, I want to provision new joiners (contractors) so that they have a standard account.
- As a line manager, I want to remove an existing user's account so that they are unable to access systems or data once they have left the company.
- As a line manager, I want to create user roles that I can assign to my employees and contractors so that it is easier to administer individual systems access.
- As a line manager, I want to assign a role to an employee so that they can access any systems associated with that role.
- As a line manager, I want to remove a role from my employee so that they can no longer access any systems associated with that role.
- As a line manager, I want to search for a user (employee or contractor) so that I can change their user roles.
- As a line manager, I want the search user feature to respond within 2 seconds so that I do not have to wait. (Example of a nonfunctional user story)

Acceptance criteria are vital to help the product owner and team have a better understanding of the general idea and limitations of both the epics and user stories. Acceptance criteria should be captured within the epic and user story, kept high level and succinct, otherwise there is a danger they will duplicate the content of the actual tests.

Example Acceptance Criteria for the Epic: "As a line manager, I want to control who can access our systems so that we ensure only authorized and authenticated individuals are permitted."

Acceptance Criteria:

- Can distinguish employees from contractors
- Can support roles for specific systems where users can be added or removed
- Must support auditing of users' sessions such as last login details

Example Acceptance Criteria for User Story: “As a line manager, I want to provision new joiners (employees), so that they have a standard account.”

Acceptance Criteria:

- Needs to capture just employee name, office location, and phone number
- Does not need to capture PPI (Personal and Private Information) as that is in the human resources systems

When constructing epics and stories and their acceptance criteria, the same considerations for attributes of a traditional waterfall/linear set of requirements should be considered in terms of data, interfaces, environment, performance, availability, regulatory, maintenance, data migration, and security requirements. With Agile there are techniques that can be adopted to help check for good attributes of stories such as “INVEST” [29] where:

- **I** – Independent
- **N** – Negotiable
- **V** – Valuable
- **E** – Estimable
- **S** – Small
- **T** – Testable

The product backlog needs to be kept up to date, reflecting and anticipating changes. The process of ensuring this is known as **backlog refinement.** Although this process is run and managed by the product owner, the work itself requires input from the development team and therefore is usually done in a small team meeting, including estimating the effort required to complete a product backlog task.

Refining the product backlog usually consists of:

- Preparing PBIs for the coming iteration – splitting items that are too large or assigning estimates or acceptance criteria to stories without them
- Making changes to existing stories – either changing estimates in the light of actual performance or reassessing their priorities
- Checking that the assessment of priorities is still correct and changing it as necessary, including the assessment of any dependencies between backlog items
- Creating new stories where a demonstration or feedback has suggested a new need
- Removing stories that no longer appear necessary

Benefits of refining the product backlog:

- A product backlog is not created with the aim of being an entire and accurate representation of requirements. It is expected that it should change during the project. Refining the backlog is the formal process of doing this. It enables development to begin while some ideas are vague – they can then be enhanced or broken down during refining, once the product owner and team have learned more during the first few iterations.
- A carefully refined product backlog will make sprint planning meetings much faster and more efficient.

- Refining stops the product backlog from becoming an endless list, and helps the team focus on completing what is necessary. This helps curb complexity, as well as budget and time overrun.

- By checking the priority formally, the product owner ensures the backlog stays up to date with changing circumstances, including what the team learned in the last iteration.

- The product owner also remains aware of items "buried" in the backlog, especially those that carry technical risk, and can review them with the team well in advance of the actual sprint.

- The product owner is always accountable for the product and sprint backlogs.

Before a sprint can start, the sprint planning session is held, where the product owner discusses with the team what is most important to work on next to meet the sprint goal. The team and the product owner discuss each potential sprint story looking to understand any architectural, testing, or development tasks, issues, or risks (including development/design and business risks including regulatory/compliance). At the end of the **sprint planning session,** the team will have committed to achieving the sprint goal, and have an **approved** and **prioritized** sprint backlog.

3.4.3 GxP/Compliance Requirements

A typical GxP regulated system often has functional requirements that have GxP impact along with nonfunctional requirements required to be in place to support GxP regulations. An example is a GxP functional requirement that includes audit trailing changes to batch recipes and approval by a compliant GxP regulated signature, with nonfunctional requirements around system security to help ensure the integrity of records within the system.

The regulated company should identify all GxP-related functional and nonfunctional requirements based on the intended use and ensure that these are included within the MVP for the initial release of the system into production use. This could include mapping those requirements against the Agile artifacts of already developed software.

Identification of requirements in terms of their potential GxP impact using tools is very advantageous. It facilitates risk-based testing focusing on these areas and ensures traceability to testing/verification activities to confirm that the MVP has been achieved from the product backlog.

Section 2.3.1 Risk Management highlights the importance of breaking a system into its functions/features and assigning a risk score to these features. This facilitates risk-based testing, where the appropriate level of rigor and detail is applied based on the level of risk, the objective and type of testing, and where it occurs in the life cycle. Within Agile, feature risk could be applied at the epic level. In addition, Section 2.3.3 Requirements indicates that it is important to consider hierarchical relationships when applying critical thinking to avoid the potential cascade of higher-risk scores to lower-level items that are actually lower risk.

For nonfunctional requirements that tend to be standard features (e.g., security), it may be beneficial to develop standard libraries of appropriate PBIs that can then be reused for subsequent software development.

Demonstrating that a system functions according to its intended use is a basic GxP requirement. It is largely demonstrated with waterfall/linear models through layers of testing activities (e.g., unit, module, integration, system) ultimately leading to a set of user acceptance tests.

For Agile, testing is always performed within the sprint. Sprints are from Scrum and Scrum says that there must be a "potentially shippable product at the end of each sprint." This means it must be fully tested, because an untested product is not shippable.

Agile testing generally uses a combination of exploratory testing and test automation with regression testing to verify that sprint developments do not impact correct functioning of software developed in previous sprints. Other test practices include solution walk-throughs or demonstrations where the potential solution is demonstrated to business users early so that they can visualize the system and identify potential issues at an early stage. Any final acceptance testing should be minimized, and test activities and records are managed and stored within tools (see Section 3.5) and referenced against the MVP and DoD. Testing that is defined within sprints should consider any requirements for regression testing scope to verify that developed and working software has not been impacted.

The fact that Agile is about rapid change may seem to present a challenge in the context of GxP and maintaining a state of control. However, this is where Agile, when operated correctly, shows its strength. In order to deliver working software and deal with rapid change, the Agile processes need to be robust, well managed, under control with joint team accountability, and with the emphasis on using tools to provide that control/oversight.

3.4.4 Approvals

As described, both the backlog refinement and sprint planning sessions are key **governance** and **approval** steps to ensure "we build the right system," and the roles and responsibilities for approvals should be defined as part of the planning.

There are many ways teams document evidence of the sprint backlog approval. Inefficient approaches often result in duplicating information, for example:

- Before the sprint can start, exporting the tool-based sprint backlog into a URS document that is then approved to indicate that the requirements for the sprint are ready to work on

- Exporting and approving a URS after multiple sprints, when ready to perform final end to end acceptance testing or User Acceptance Testing (UAT) of the release

These approaches to creating a document version of the product backlog are merely snapshots in time and duplicate information. These duplications are extremely prone to becoming out of date with the actual product backlog, tests, and code as time moves on. **Having a single source of truth is more important than the perceived need for a traditional document.**

3.4.5 Acceptance Instead of Approval

There is a misconception that approval means sign, and by extension that if using a tool/electronic system, then there is the need to ensure any electronic approvals/signatures comply with the pharmaceutical regulatory requirements on electronic signatures, for example 21 CFR Part 11 (US) [30]. This is not the case, and is only applied where the approval is a predicated signature/approval by regulation. For software life cycle deliverables, this is only the case for Software as a Medical Device (SaMD) or software embedded in a regulated medical device. Approval can also be achieved by other means such as status change, email, audit trails.

Some tools have immutable change logs built into them. These logs record exactly *who,* did *what,* and *when* against each PBI. Plus, these tools can be configured to enforce security rights over exactly who can do what.

The tools also have a start-sprint concept that officially starts the clock of the sprint, which is recorded within the logs.

These logs are the record of the sprint backlog (plan), and a record of what PBIs (requirements) are **approved** to be worked upon during the sprint. Teams use this record in place of a signature on a URS document.

During the sprint, PBIs are worked upon and typically moved from a To Do state, to an In Progress state, and finally to a Done state, once the team and the product owner are satisfied the individual PBI is finished, with reference to the MVP and DoD or equivalent as a control point to confirm that the required criteria have been met.

By limiting who can change the status of a PBI to Done to only the product owner (or an authorized team member, e.g., quality manager), teams are also able to show a record of evidence that the PBI has been satisfactorily completed, works as specified, and is sufficiently tested, and therefore, **accepted** by the product owner.

Do not be afraid to use tools. The pharmaceutical industry, regulators, etc., have agreed for many years that tools and systems are the sensible way to maintain and manage GxP records and data related to the pharmaceutical product life cycle, and that paper documents and records are typically not effective.

Records should be maintained because they are of **value to the organization,** rather than for a regulator or external auditor. Note that the FDA CDRH Case for Quality program [5] encourages the use of tools.

> *"The work that you do should be valuable to the organization, right? You're maintaining a record. You're maintaining the activities, not necessarily because you need to demonstrate it to the Agency or to any other Auditor. You're doing this work and maintaining this record because it becomes your source of truth for your organization down the road."*
>
> **Cisco Vicenty, CDRH FDA** [31]

When it comes to inspections, while it might be uncomfortable for some organizations to use a tool to document requirements, it is worth noting that companies (including pharmaceuticals) in high-risk industries like medical devices, defense, telecommunications, and power generation and distribution have controlled requirements in requirement management tools instead of documents over the last two decades. Tool usage is not new, but how organizations handle changing requirements is, and they need to be prepared to show inspectors the records within the tools as part of regulatory inspections.

3.5 Tools Instead of Documents

There are numerous advantages to utilizing tools for fully realizing the benefits of Agile, and many benefits of Agile cannot be achieved without their use. Agile can provide opportunities for an improved software development process but trying to apply or overlay the traditional documentation approach presents a significant barrier, potentially introducing risks and noncompliances should discrepancies occur between documentation and the primary records in the tools. Tools allow teams to address the compliance and quality requirements in an efficient way that does not detract from the overall value that the Agile process brings.

From a GxP perspective, it is very important to understand the scope and role of the software life cycle management tools used as part of Agile software development. If the goal is to demonstrate that the system is fit for purpose, and that all functionality can be traced back to requirements, and prove that testing was done against those requirements, then it is necessary to look across multiple tools and understand how they interact.

There are many benefits justifying the use of Agile for software development such as:

- Providing opportunities to assess the direction of a project throughout the development life cycle
- Allowing teams an opportunity to learn with each new iteration
- Providing the ability to make quick course corrections based on stakeholder feedback
- Realizing faster delivery of an operational system based on the defined MVP
- Empowering teams to work creatively and effectively

- Defining and elaborating on requirements just-in-time so that knowledge of product features is as relevant as possible

- Improving quality because testing starts from day one: incorporating continuous integration and daily testing into the development process

Many organizations use Agile and have discovered that the utilization of tools enables full realization of benefits. A comprehensive tool set can reduce the burden and overhead of documentation, provide elements of automation, and allow a team to be nimble as Agile intended. The required investment in a full suite of tools may be significant, but this investment needs to be set against the efficiencies and automation that tools can provide at an increased speed of delivery.

Agile is suited for the utilization of tools because of the way in which the process is fundamentally designed. For example, the management of user stories in a product backlog that can be continually refined and updated is a core strength of Agile. This is one of the most powerful aspects of Agile, which allows it to not only handle changing or emerging requirements, but also improve the solution because of those changes. A backlog with continuously changing user stories is extremely difficult to manage on paper; therefore, demonstrating control (a key principle of GxP) is equally difficult. A tool can provide an efficient and effective means to manage user stories in a backlog throughout updates and changes.

While backlog management is the primary activity where tools are utilized, there are other areas as well. Tools can manage a variety of Agile activities like user story risk assessments, testing, and traceability. A fully comprehensive toolset can provide an integrated solution to manage them all. With tools in use the need for documentation will in many cases be eliminated or, at a minimum, be significantly reduced, as well as eliminating risks of transcription errors or versioning or content anomalies where parallel electronic and documentation-based systems are in use.

3.5.1 Benefits of Using Tools

What are some of the benefits of using tools with Agile? Probably the biggest benefit is the speed at which updates can be made and accepted versus times for documentation to be updated and approved.

A good example is backlog management. Some of the basic tenets of Agile are the ability to quickly respond to change and elaborate on requirements just-in-time. Compare this with approved Product Backlogs that need to be revised and approved after each user story update. With the use of a backlog management tool, user stories can be individually updated and approved without revising and approving large requirements documents.

Reports may also be generated from tools, providing baseline status at points in time (for example, of user stories), and used, for example, as an aid during regulatory inspections to provide evidence of status.

Another example is providing traceability from requirements to testing, which is a requirement for GxP system projects. Traditionally traceability is demonstrated by separate traceability matrix documents created as an additional layer of documentation that needs to be manually updated as the project progresses. A tool can provide embedded, automatic traceability without the need to create and manually maintain a separate trace matrix document. Instead, traceability can be dynamically achieved and demonstrated by the tool, which, if necessary, can generate reports at key stages, such as at the completion of testing.

A not so readily apparent area where tools provide benefit is in driving quality improvement. Tools can reduce the chance for human error when utilized correctly.

Testing is a great example of where tools provide opportunities for improved quality, providing for a higher volume of testing with more focus on expected functionality. Behavior-Driven Development (BDD) is the product of an evolution of user story development that allows users to write stories not only in natural language but also in a way in which user stories can be directly translated into test cases. This produces testing that is appropriately focused and with automation, provides the capability of a high volume of quickly executed test scripts with less resource impact to the project team. Not only can automated testing be executed much more quickly compared to manual testing, but the scope and coverage of testing can be greater because of the ability to create a higher volume of test scripts.

3.5.2 Types of Tools

Tools that support the journey to Agile are the ones that support the principles of Agile. The 2020 COVID-19 pandemic, when individuals were unable to directly interact face-to-face to deliver Agile projects, illustrated how tools are an indispensable resource to a Scrum team.

The principles of the Agile Manifesto [23] are clear, but what has been seen in practice is that tools work in conjunction with the principles. There is no conflict when the emphasis is placed on tools; in fact, teams can increase their output and quality. This is most beneficial for larger teams that want to mature their Agile practice, but still provides benefit for smaller teams beginning to adopt Agile.

Tools in an Agile Journey:

Table 3.1 is a representative listing of tools in common use at the time of this publication but is not to be regarded as exhaustive. New and enhanced tools continue to be developed and used over time.

Table 3.1: Common Agile Tools

Tool	What It Is	What It Does	Agile Principles	Quality
Backlog Management Software	This software consists of a versatile tool set. It can track many things from bugs to resources and multiple Scrum teams' project work using workflow, visual boards, Kanban, and reports. This software usually contains multiple application programming interfaces for a better user experience across multiple tools to provide high-level dashboards.	Helps organize issues, assign work, and follow team activity from initial intake to development to release. With extensive plugins, additional work such as testing and design may be managed. It makes obstacles to resolving issues transparent and simplifies collaboration.	Allows teams to **collaborate** across functional boundaries on many issues using Scrum tools like Kanban boards, Scrum boards, backlog management, reports and dashboards, etc. **Responding to change –** Backlog Management Software manages the updates to user stories, including traceability and acceptance of such changes. This means the team can focus on the product they are developing, rather than the process of manually documenting. The responsive nature of Backlog Management Software gives teams flexibility to adapt to the user's changing needs.	Provides real-time status of requirements including traceability (e.g., to testing), and approval status (e.g., accepted, done), and includes an audit trail for changes and reporting functionality.

Table 3.1: Common Agile Tools (continued)

Tool	What It Is	What It Does	Agile Principles	Quality
Testing Management	This may be a stand-alone application or a plug-in for other software suites. It manages all aspects of testing from test creation to execution and review and approval.	Allows teams to automate (it should work manually as well) some basic Software Quality Assurance. The tedious task of tracking coverage can be done through visual cues so when a story is created it is possible to see if there is a test case created. Provides dashboards to view total coverage as well as traceability to user stories and to code. A standard set of tests can be made available across teams. Through integrations with orchestration software, teams can run them as part of code commits to the source code repository so automated tests can run prior to code merge.	Enables working software through standard processes. Aids in proactive test set management so tests can easily and automatically verify code for a quicker delivery. If configured, it can run a consistent set of tests for each delivery through automation, without affecting the speed so working software is delivered sooner. A standard set of tests ensures an acceptable level of quality each time. Allows developers to fail faster so they can self-correct without waiting for demos. A good tool for new tests to leverage best practices, which includes writing scenarios using Behavior-Driven Development (BDD) in the Gherkin-based framework.[8] Although individual interactions are preferred over tools and processes up-front when gathering requirements and during demos, users want to avoid multiple hand-offs during the testing phase. Testing software's great advantage is speed of feedback, which is further realized through integrations with standard testing automation frameworks (i.e., Cucumber, Robot Framework, Selenium).	Provides evidence of test status, using automation and regression test functionality to provide additional assurance of thorough testing. Can provide functionality to perform negative and stress testing, and reduces the likelihood of test script errors that occur when tests are developed manually.

[8] Gherkin syntax is a method of developing test cases using plain language in the "Given, When, Then" format against scenarios/features.

Table 3.1: Common Agile Tools (continued)

Tool	What It Is	What It Does	Agile Principles	Quality
Orchestration Software	Provides the ability to build and deploy software code in a given sequence from specified locations.	It is used in conjunction with code repositories and large binary registries to control and orchestrate software deployment with other tools like Chef, Ansible, and Puppet.	Like many of the tools, Orchestration software enables working software through standard processes. It is a key part of Continuous Integration and Continuous Deployment (see Section 3.6). It automates many of the routine tasks through automated pipelines that build software in a prescriptive manner, so software to be tested is always tested on a stack that is configured and built the same way each time. This removes errors introduced in the process of misconfiguration (or detects them) and, for example, should reduce the risk of unfinished PBIs from being deployed. Pipelines are built with quality checks to ensure working software is delivered rather than concentrating on comprehensive documentation. Often the documentation is to ensure there is quality.	Provides automated assurance of working and tested code, and defined workflow for the code moving from environments.
Code Repository Software	Distributed version control and code management system. It makes it easy for teams and multiple developers to collaborate within the same source code and maintain control when creating software.	Allows approval of code review more efficiently when code is committed to the repository. Enables parallel software versions to be managed and controlled.	**Individual interactions** are key for good code development. Code Repository Software allows a team to expose potential bugs and raise discussions right in the source code. If integrated with Backlog Management Software and orchestration software, it provides traceability from requirements down to the code level. Teams can perform documented code reviews based on a defined quality workflow to ensure consistency across multiple teams and release working software sooner.	Reduces the chances of configuration management issues particularly where multiple developers may be working on the same code, and enables traceability from requirements to code level automatically.

Table 3.1: Common Agile Tools (continued)

Tool	What It Is	What It Does	Agile Principles	Quality
Large Binary File Registry	Provides artifacts storage, version control, and secure access of software artifacts (repository for code and binary files).	Allows for a secure authenticated way to store and retrieve software.	**Working software** over complex and comprehensive documentation for binary file management. Having multiple teams and projects often means multiple storage repositories for software assets. The Binary File Registry works with application program interfaces to allow universal repository access and management of assets. This removes the need for working with different software package management systems. It enables teams to provide consistency in Continuous Integration and Continuous Deployment workflows.	Enables accurate configuration management and reduces complexity where multiple repositories are required.
Knowledge Management Software	Provides virtual workspace. Spaces help organize content into meaningful categories, like different folders to store technical documents.	Gives each function their own space so they can focus and make their information easier for everyone to find. Used for collaboration when drafting a document. If using for regulated documents, then normal document control system controls should be applied for managing controlled deliverables.	**Customer collaboration** as well as team collaboration is the key principle that knowledge sharing software enables. With the proliferation of multiple SharePoint sites and other such tools, it is difficult for teams to find the necessary information for all of their project work in one place. With integrations into backlog tools and code repositories, it facilitates individual interactions. The power of this tool is in asynchronous collaboration as times and work arrangements for team members may vary greatly.	Assists team members in locating information quickly and accurately.

3.5.3 Controls

No examination of tools would be complete without considering compliance requirements/controls. Some project teams have historically avoided the use of tools because of the uncertainty and overhead associated with the perceived validation requirements. The reality is that the use of tools brings so much value, that avoiding their use due to questions and concerns about validation, is highly detrimental.

Regardless of the development method used, the software life cycle must be formalized, including tool selection and use. The approach to maintaining records/information needs to be defined and controlled, including traceability of information across tools (for example, from backlog management into test tools) and any requirement to produce reports from tools.

With appropriate, carefully selected and integrated tools that are effectively configured and used by trained users in an environment of robust user and privilege management, the tools provide much greater support to control the entire development life cycle, likely greater than with a traditional document-based approach.

The type of controls required for tools requires an understanding of the intended use and willingness to implement controls that are commensurate with risk. FDA CDRH has signaled that it does not consider tools that support the testing of automation or systems used in production or as part of the quality system to be directly part of that production or the quality system. [5] They have also indicated that the activities performed and the records maintained for these supporting tools is the responsibility of the manufacturer and they will not be viewed as requiring the same rigor with respect to controls as systems that directly support production or the quality system. [31]

Within *ISPE GAMP 5* [2], the tools used as part of Agile are covered within the scope of Appendix M4, Section 3.1 and classed as Category 1 – Infrastructure Software, especially as the tools used are generally in broad use throughout the software industry. There is, however, an additional consideration in that information that previously was stored as documentation is now solely stored within the tools. Consider, for example, acceptance test results stored within a tool. The results need to be trustworthy and available, and therefore controls should be defined and applied to ensure the records are adequately protected, retained, and available.

The key to understanding the control requirements for tools is in their intended use. A tool used to manage only the backlog requires fewer controls than a tool used to manage and document testing of high-risk software.[9] This is the same as a tool with built-in functionality for design control traceability requiring a different level of control if it represents the official traceability for a medical device.

Each organization, and in some cases project teams, need to assess how they use tools including any dependencies/ interfaces between tools, the record retention times and controls required, and how to implement appropriate controls for those tools based on their risk tolerance. Documentation associated with ensuring tool fitness for use should not be an impediment. However, the checks relied upon to implement quality controls, for example, as part of configured workflows, should be verified to ensure they are working as intended. This approach is supported in the case of testing and tools by EU Annex 11 Section 4.7 [19], which states that:

> *"Automated testing tools and test environments should have documented assessments for their adequacy."*

ISPE GAMP RDI Good Practice Guide: Data Integrity by Design Appendix S2 Computer Software Assurance [6], notes:

> *"that the use of incidental tools to aid in the validation effort does not trigger a separate validation effort; automated tools only require a documented assessment for their adequacy."*

Such a tool is not a GxP regulated system, and its acceptability should be documented using a company's non GxP-business practices, for example, by the application of good engineering practices and evidence of proper selection, installation, and control.

[9] High-risk software is software that could directly impact patient safety, product quality, or related data integrity.

3.6 DevOps, Continuous Integration/Deployment, and Product Teams

In a DevOps environment, the boundary between product development and product maintenance teams is eliminated and a single team works on both development and support. The benefits of this are enhanced team spirit where focus can be placed on important/urgent tasks, avoiding siloes in the team, and enabling team members to develop through a broader variability/experience of activities.

Experience shows that this approach can result in higher velocity, where velocity is work completed within a given time frame (for example, the number of epics completed within a sprint), and after some months, a better working climate, and higher end-user satisfaction.

Applying DevOps to the final configured computerized system successfully requires reliance on tools and the information within these as the higher change frequency is unlikely to be achievable using traditional documentation and the need to update/maintain.

In a DevOps environment, systems will undergo many changes/releases, which initially seems to be difficult to reconcile in a GxP regulated environment. This concern can be mitigated with the level of control and oversight of the process, together with the tools and accountability of the DevOps team for the quality of the product. However, there is still a level of risk/impact assessment required that will reveal potentially higher-risk GxP functions undergoing change, and therefore drive more controls/assurance through the change.

Continuous Integration (CI) and Continuous Deployment (CD) (sometimes also referred to as continuous delivery) is an extension of DevOps where there is increased use/reliance on automated software analysis tools. Code is delivered into code repositories and is verified (tested) and integrated using these tools. The continuous delivery/deployment process ensures that the code delivered by the CI process is suitable for deployment. The distinction between continuous delivery and continuous deployment is that there is still human oversight/approval to the final deployment to production in continuous delivery, whereas the increased use of tools with continuous deployment replaces the final human oversight step with tools.

Additionally, DevOps requires business buy-in from process owners and business QA groups as their input/expertise is likely to be called upon more frequently than with traditional change/release updates.

DevOps can be thought about more as focusing on culture and roles, whereas CI/CD is about practices of applying tools and automation.

3.6.1 Quality and Regulatory Aspects

Adherence to the defined process (for example Scrum ceremonies) helps to drive quality, ensures timelines can be adjusted quickly, and provides a focus on learning by addressing issues as they appear.

DevOps teams provide CI and continuous delivery with faster turnaround times for increments (more velocity).

The use of tools and records and discipline in the DevOps team following the process drives better quality.

DevOps provides for closer integration of the product and the associated tool sets that in turn provide additional controls such as:

- No development before an accepted user story
- Source code review and automated testing as part of the CI process to help avoid the release of bad code
- Ensuring close alignment and constant engagement with the Product Owner as feedback on proposals comes almost instantly

Also, for the DevOps team there is a greater perspective on the long-term development/evolution and support of the product rather than the development team just focusing on annual releases and maintenance teams supporting what they are given.

It is beneficial if DevOps teams feel that they have accountability within the team for compliance. If compliance is perceived as something outside, then the team sticks together and stakeholders outside such as security, compliance, and auditors are perceived as "strangers."

3.6.2 Inspections

Successful inspections can come from two key factors where DevOps and product teams have an advantage:

- Do what you say and say what you do
- Correct errors and make the right product fit for purpose

Having the right team is pivotal, and good collaboration is essential, so smaller team sizes tend to be more effective. [32] Having the competencies (including compliance) in the team so that it can work independently is very motivating. Having a trustful and respectful atmosphere is important, as is being open about strengths and weaknesses so they can be addressed; input from sprint review (improvement sessions) must be taken seriously.

4 IT Service Management

4.1 Introduction

Regulated companies are increasingly utilizing IT/IS (Information Technology/Information System) service companies to provide IT infrastructure, software, and data services. Different models are used to deliver these services to regulated companies including traditional outsourcing models to Infrastructure, Platform, and Software "as a Service" (collectively referred to as XaaS or cloud computing). Even within these broad service model definitions there are significant variants. For example, SaaS can relate to a single instance, multitenant offering (i.e., multiple regulated companies sharing access to the same software platform with segregated data) with limited customer configuration and opportunity for customers to conduct their own validation. It can also mean a single instance, single tenant, highly configurable solutions (i.e., regulated companies have their own instance of the application) that may require significant validation effort from the regulated company.

Regulated companies leverage IT/IS service provider effort in support of their regulatory accountabilities. Regulated companies must establish a risk-based assessment and governance strategy to ensure that the IT/IS service provider adheres to IT, quality, and information security controls that ensure services and solutions are fit for intended use and data integrity is maintained.

Responsibility for IT-related activities may be delegated to IT/IS service providers, but in all cases regulatory accountability lies with the regulated company. When outsourcing or delegating activities, there should be no resultant decrease in product quality, process control, or QA. There should be no increase in the overall risk to the GxP processes. The competence and reliability of service providers must be ensured.

IT/IS service providers should adhere to industry guidance, standards, and practices when establishing the IT QMS (e.g., ITIL® Foundation: ITIL 4 Edition [33], COBIT® [34], ISO 9001 [35], *ISPE GAMP 5* [2], etc.). Additionally, information security standards (e.g., ISO 27001 [36], NIST – US National Institute of Standards and Technology [37], Cloud Security Alliance [38]), and other requirements such as the Sarbanes-Oxley Act of 2002 (US) [39] and data privacy influence a set of integrated IT controls. Adherence to these cross-industry practices supports the creation of a robust controls framework that ensures IT/IS services are provided in a controlled manner fit for intended use.

IT/IS service providers often utilize electronic systems such as configuration management tools, service management tools, application life cycle management tools, and automation in support of IT processes and controls. The use of such tools and automation is strongly encouraged, as it supports process adherence, enforces process control points, and ensures that relevant data is captured as required.

This chapter clarifies good practices related to the assessment, governance, and leveraging of IT/IS service provider knowledge, experience, and artifacts, particularly those related to XaaS.

4.2 Scope

This chapter defines and clarifies:

- Different IT service models and associated risks to regulated user companies
- Relationships between a regulated company and IT/IS service provider
- Requirements of the Pharmaceutical Quality System (PQS) QMS and IT QMS
- Approaches to IT/IS service provider assessment and governance
- Approaches to management of IT infrastructure

- Approaches to validation of SaaS
- Approaches to data governance within XaaS
- Leveraging service provider knowledge, experience, and artifacts in support of the regulated company GxP activities

4.3 Accountabilities and Responsibilities of Regulated Companies and IT/IS Service Providers

Regulated companies are accountable for compliance with GxP regulations. IT/IS service providers are responsible for establishing IT quality management and IT information security controls that govern the services provided to the regulated company. Regulated companies must assess IT/IS service providers, in accordance with risk, to ensure that their controls can be relied on by the regulated company in support of their regulatory accountabilities.

When entering into a contract with an IT/IS service provider, the regulated company needs to consider:

- Requirements for risk-based assessment of the IT/IS service provider (including periodic assessments and for-cause assessments) to ensure that appropriate quality and information security controls are in place
- Master Service Agreements (MSAs) and/or SLAs defining the services, roles, and metrics used to monitor service performance and quality
- MSAs, SLAs, and quality agreements defining the quality management expectations in order for the regulated company to leverage IT/IS service provider effort in support of the regulated company's GxP accountabilities
- Clearly defining escalation paths for timely resolution of serious issues
- Clear communication channels between the regulated company and IT/IS service provider to notify of incidents, release plans, etc.

4.4 Leveraging Supplier Effort

Throughout this chapter, reference is made to leveraging IT/IS service provider effort. By effectively leveraging this effort, the regulated company maximizes value and avoids duplication of effort by ensuring activities are undertaken by organizations with the prerequisite skills and experience.

Leveraging is enabled through a risk-based approach to IT/IS service provider assurance that ensures service providers adopt the necessary controls and records within their QMS. This demonstrates that services and solutions are fit for intended use and data integrity is maintained.

Leveraging does not imply that service providers must provide the regulated company with documentation and records as defined by, and part of, the regulated company's QMS. The value of service provider documentation and records is within the service provider's QMS where they support and demonstrate the effectiveness of a service provider's processes and controls. This does not preclude service providers from providing copies of documentation and records where this facilitates an effective and efficient approach to assurance.

Regulated companies need to apply critical thinking during the assessment of service providers and be familiar with current approaches to IT/IS service and solution delivery.

The regulated company and service provider should agree on the effective assurance approach based on the risk of services and solutions being provided.

4.5 IT Service Quality Management

4.5.1 Pharmaceutical Quality System versus IT Quality Management

There are some key differences between the regulated company's PQS and the service providers IT quality system.

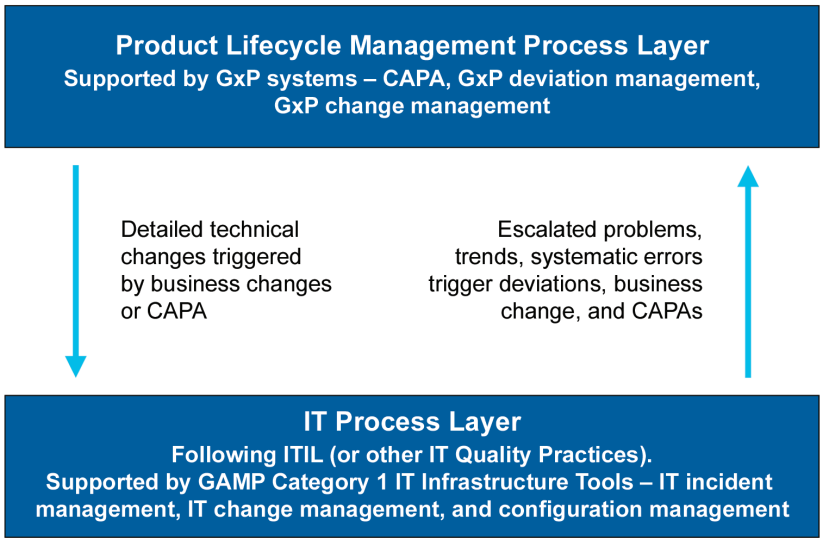
The PQS supporting the medicinal product life cycle includes processes such as deviation management, CAPA, and change management. These GxP processes engage business and QA SMEs to effectively evaluate and manage risks impacting patient safety and product quality.

Similar processes are included in the IT QMS, but they focus on IT risks relating to availability, performance, and information security. These processes include incident, problem, configuration, and change management that engage IT SMEs in the evaluation and management of IT risks.

Where IT/IS services are provided by organizations within the regulated company, IT/IS processes may be managed within the overall company quality system, but the relationship between the IT/IS processes and the PQS processes is the same.

Figure 4.1 demonstrates the interfaces established between the IT QMS and the product life cycle QMS to hand off potential risks relating to patient safety and product quality. For example, an IT Incident may potentially impact GxP data (e.g., data loss or corruption). Such incidents should be evaluated and communicated in accordance with risk, to business and QA functions so that the business and regulatory impacts can be assessed and mitigated. The Configuration Management Database (CMDB) can be effectively used to identify GxP-relevant IT infrastructure assets, applications, and data, providing the link between IT/IS processes and the PQS processes.

Figure 4.1: Relationship between Product Life Cycle Quality System and IT Quality System



4.5.2 IT Quality Management System

An IT QMS based on well-established, cross-industry IT governance standards such as the ITIL Foundation: ITIL 4 Edition [33] is essential to the effective management of IT services. The IT QMS establishes robust processes, technologies, and subject matter expertise to effectively manage the IT infrastructure environment in accordance with risk. Further, such IT QMS use metric-drive continual improvements to enhance IT controls maturity. [40]

Examples of IT practices (as prescribed by ITIL [33]) include:

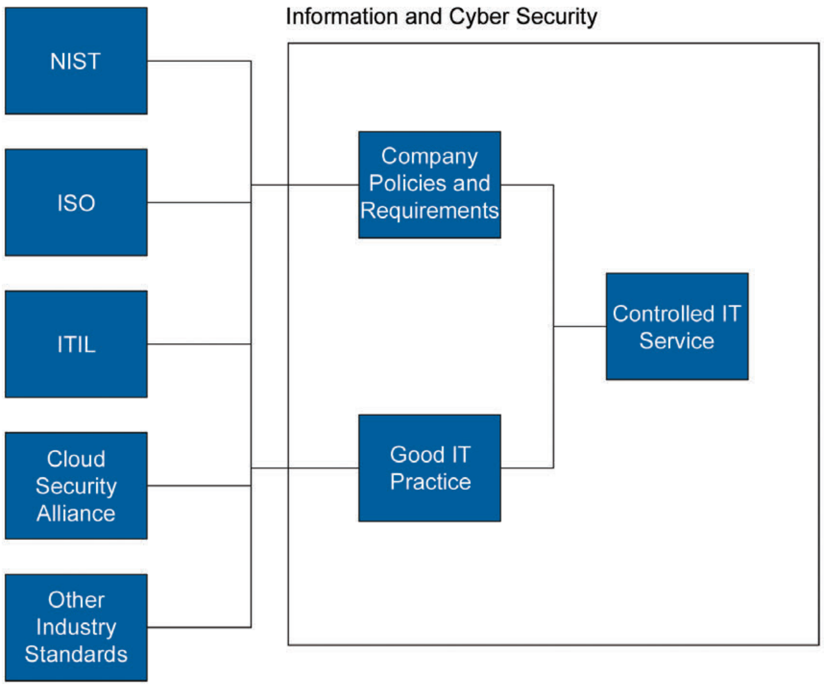
- General management practices:
  - Risk Management
  - Knowledge Management, Skills Development and Training
  - Measurement, Performance and Continuous Improvement
  - Customer Relationship, Supplier Management
  - Document and Records Management
- Service management practices:
  - Incident, Problem and Service Request Management
  - Service design, configuration, validation and testing
  - Service level management
  - Availability, capacity, performance, and continuity
  - Monitoring and event management
  - Release management
  - Change management
- Technical management practices:
  - Deployment management
  - Infrastructure and platform management
  - Software development and management

Other QMS and information management system standards such as COBIT [34], ISO 27001 [36], ISO 9001 [35], NIST [37] further define the IT controls employed by IT/IS service providers.

4.5.3 Synergies between GxP and Other Industry Regulatory Requirements

Information security and cyber security are crucial to ensure data integrity. Regulated companies must ensure that appropriate procedural and technical controls are in place to ensure data integrity is maintained throughout the data life cycle. Regulated companies need to leverage cross-industry information security standards, for example, relating to cyber security (see Figure 4.2), so that appropriate controls are in place, as GxP regulations and guidance do not specifically address information security best practices.

Figure 4.2: Influence of Industry Good Practices on IT Quality and Security System



Regulated companies remain accountable for data integrity and need to assess Infrastructure as a Service (IaaS), Platform as a Service (PaaS), and SaaS providers to ensure such controls are in place. This requires collaboration between multiple organizations within the regulated company including Quality, IT, information security, and possibly other organizations such as data privacy.

It is therefore beneficial to look at where standards and regulations in the two domains of quality and security can complement each other and where gaps exist. This is essential for the IT/IS service provider assessment process, system development, and in relation to the continuous operation.

4.5.3.1 Quality and Information Security Synergies

When computerized systems are validated for their intended use, it is not only done with quality in mind. As more IT/IS services are implemented and integrated in regulated companies, requirements for information security are increasing. Table 4.1 shows that quality and information security fit well together and must be handled accordingly.

Security is broadly referenced within regulations such as 21 CFR Part 11 [30] and EU GMP Annex 11 [19]. Development of common standards supporting data integrity has improved in many industries and is reinforced by privacy regulations, for example the EU General Data Protection Regulation (GDPR) [41] and California Consumer Privacy Act (CCPA) [42]. In relation to security standards, regulated companies can benefit from many industry standards such as ISO27001 [36] and NIST [37]. However, there is a difference between a quality approach and a security approach, and it is important to be aware of this when establishing IT processes. For example, requirements regarding management evaluation of QMS efficiency are not included in ISO 27001. Table 4.1 outlines the cross references between NIST Security Framework [43], ISO 27001, and GxP regulations. Table 4.1 also describes how good engineering practices from other standards outside the pharmaceutical industry are relevant to GxP operations.

Table 4.1 provides examples of which regulations address information security requirements. These are examples only and other regulatory requirements and guidance such as EMA Guideline on computerised systems and electronic data in clinical trials[10] [44] also address key information security requirements.

[10] Draft at time of publication.

The NIST [43] model of security and privacy controls (summarized in Figure 4.3) describes the functions and categories of cyber security activities upon which Table 4.1 is based.

Figure 4.3: NIST Cybersecurity Framework [45]

NIST [43] explains this framework:

- ***"Identify –*** *Develop an organizational understanding to manage cybersecurity risk to systems, people, assets, data, and capabilities."*
- ***"Protect –*** *Develop and implement appropriate safeguards to ensure delivery of critical services."*
- ***"Detect –*** *Develop and implement appropriate activities to identify the occurrence of a cybersecurity event."*
- ***"Respond –*** *Develop and implement appropriate activities to take action regarding a detected cybersecurity incident."*
- ***"Recover –*** *Develop and implement appropriate activities to maintain plans for resilience and to restore any capabilities or services that were impaired due to a cybersecurity incident."*

Table 4.1: Alignment of GxP Requirements and Information Security Standards/Industry Guidance

Identify	
Asset Management: The data, personnel, devices, systems, and facilities that enable the organization to achieve business purposes are identified and managed consistent with their relative importance to business objectives and the organization's risk strategy.	**Example Regulatory Requirements:** FDA 21 CFR 211.25, 211.28, 211.63, 211.67, and 211.68 EU GMP Annex 11, §1 and §4.3 **Example Information Security Standards/Industry Guidance:** ISO 27001 A.8 Asset Management
Business Environment: The organization's mission, objectives, stakeholders, and activities are understood and prioritized. This information is used to inform cybersecurity roles, responsibilities, and risk-management decisions.	**Example Regulatory Requirements:** FDA 21 CFR 211.25, 211.28 and 211.180 EU GMP Annex 11, §1 and §2 **Example Information Security Standards/Industry Guidance:** ISO 27001 Sec 4 Context of the Organization
Governance: The policies, procedures, and processes to manage and monitor the organization's regulatory, legal, risk, environmental, and operational requirements are understood and inform the management of cybersecurity risk.	**Example Regulatory Requirements:** FDA 21 CFR 211.25 and 211.28 EU GMP Annex 11, §1, §4.6 and §10 **Example Information Security Standards/Industry Guidance:** *ISPE GAMP 5,* Chapters 2, 3, and 4 ISO 27001 Sec 4.3 Determining the scope of the information security management system
Risk Assessment: The organization understands the cybersecurity risk to organizational operations (including mission, functions, image, or reputation), organizational assets, and individuals.	**Example Regulatory Requirements:** FDA 21 CFR 820.30 EU GMP Annex 11, §1 **Example Information Security Standards/Industry Guidance:** *ISPE GAMP 5,* Appendix M3 ISO 27001 Sec 6.1 Actions to address risk and opportunities
Risk Management Strategy: The organization's priorities, constraints, risk tolerances, and assumptions are established and used to support operational risk decisions.	**Example Regulatory Requirements:** EU GMP Annex 11, §1 **Example Information Security Standards/Industry Guidance:** *ISPE GAMP 5,* Appendix M3 ISO 27001 Sec 6.1.2 Information security risk assessment
Protect	
Access Control: Access to assets and associated facilities is limited to authorized users, processes, or devices, and to authorized activities and transactions.	**Example Regulatory Requirements:** FDA 21 CFR 211.68; FDA 21 CFR Part 11, §11.10(d) and 11.300 EU GMP Annex 11, §2 and §12ff **Example Information Security Standards/Industry Guidance:** ISO 27001 Annex A.9 Access control
Awareness and Training: The organization's personnel and partners are provided with cybersecurity awareness education and are adequately trained to perform their information security-related duties and responsibilities consistent with related policies, procedures, and agreements.	**Example Regulatory Requirements:** FDA 21 CFR 211.25 and 211.28 EU GMP Annex 11, §2 **Example Information Security Standards/Industry Guidance:** ISO 27001 Sec 7.3 Awareness
Data Security: Information and records (data) are managed consistent with the organization's risk strategy to protect the confidentiality, integrity, and availability of information.	**Example Regulatory Requirements:** FDA 21 CFR 211.68; FDA 21 CFR Part 11, §11.10(d) EU GMP Annex 11, §1 and §12ff **Example Information Security Standards/Industry Guidance:** *ISPE GAMP 5,* Appendix M3 ISO 27001 Sec 0.1 General …*"The information security management system preserves the confidentiality, integrity, and availability of information by applying a risk management process"*

Table 4.1: Alignment of GxP Requirements and Information Security Standards/Industry Guidance (continued)

Detect	
Security Continuous Monitoring: The information system and assets are monitored at discrete intervals to identify cybersecurity events and verify the effectiveness of protective measures.	**Example Regulatory Requirements:** FDA 21 CFR part 11, §11.10f EU GMP Annex 11, §9 and §13 **Example Information Security Standards/Industry Guidance:** ISO 27001 Sec 9.1 Monitoring measurement, analysis and evaluation
Detection Processes: Detection processes and procedures are maintained and tested to ensure timely and adequate awareness of anomalous events.	**Example Regulatory Requirements:** FDA 21 CFR Part 11, §11.10f EU GMP Annex 11, §9 and §13 **Example Information Security Standards/Industry Guidance:** ISO 27001 Annex A.12.4 Logging and monitoring
Respond	
Response Planning: Response processes and procedures are executed and maintained to ensure timely response to detected cybersecurity events.	**Example Regulatory Requirements:** FDA 21 CFR 820.30 EU GMP Annex 11, §1 **Example Information Security Standards/Industry Guidance:** *ISPE GAMP 5,* Appendix M3 ISO 27001 Annex A.17.1.1 Planning information security continuity
Analysis: Analysis is conducted to ensure adequate response and support recovery activities.	**Example Regulatory Requirements:** FDA 21 CFR 820.30 EU GMP Annex 11, §1 **Example Information Security Standards/Industry Guidance:** *ISPE GAMP 5,* Appendix M3 ISO 27001 Annex A.16.1.4 Assessment of and decision on information security events
Mitigation: Activities are performed to prevent the expansion of an event, mitigate its effects, and eradicate the incident.	**Example Regulatory Requirements:** FDA 21 CFR 820.30 EU GMP Annex 11, §1 **Example Information Security Standards/Industry Guidance:** *ISPE GAMP 5,* Appendix M3 ISO 27001: Annex A.16.1.5 Response to information security incidents
Recover	
Recovery Planning: Recovery processes and procedures are executed and maintained to ensure timely restoration of systems or assets affected by cybersecurity events.	**Example Regulatory Requirements:** FDA 21 CFR 211.68 EU GMP Annex 11, §1 and §16 **Example Information Security Standards/Industry Guidance:** *ISPE GAMP 5,* Appendix M3 ISO 27001 Annex A.17.1 Information security continuity ISO 22301 Security and Resilience – Business continuity management Systems – Requirements
Improvements: Recovery planning and processes are improved by incorporating lessons learned into future activities.	**Example Regulatory Requirements:** FDA 21 CFR 211.68 EU GMP Annex 11, §1 and §16 **Example Information Security Standards/Industry Guidance:** *ISPE GAMP 5,* Appendix M3 ISO 27001 Annex A A.16.1.6 Learning from information security incidents
Notes: FDA: 21 CFR 211 [46], 21 CFR 820 [47], Part 11 [30] EU GMP: Annex 11 [19]	ISO: 27001 [36], 22301 [48] *ISPE GAMP 5* [2]

4.5.3.2 Data Security

Data security risk can be associated with loss, leakage, or unavailability of data and unintended or unrecognized alteration of data. This can cause business interruption, loss of revenue, loss of reputation, or regulatory noncompliance.

Common to IT governance frameworks, guidance and requirements including *ISPE GAMP* [49], ISO27001 [36], NIST [37], and EU GDPR [41] are the concepts of defining, assessing, treating, and managing risk. A risk register may be a helpful tool for an organization managing their more traditional application-level risks with those induced by an "as a Service" delivery. The risk register concept can also fully support the data protection impact assessment requirements of the GDPR, including the EU, Swiss [50], and UK GDPR [51], and will similarly serve as a common touchpoint for data integrity risk management.

Managing privileged access and segregation of duties is critical to securing data in the cloud. These accounts provide unlimited access to high value applications and data (e.g., medical records) and continue to be a top target for cybercriminals. (The 2019 Cloud Security Alliance report has identified *account hijacking* as a top five threat. [52]) Privileged user risk is heightened in a cloud environment because of the dynamic nature of cloud computing. For example, new virtual server instances can be provisioned rapidly at scale, which can introduce new privileged accounts into the environment (with default passwords). IT/IS service providers need to have appropriate controls in place for managing and monitoring privileged accounts.

One of the principles in *ISPE GAMP 5* [2] is to avoid duplication of activities and documentation between the IT/IS service provider and regulated company. Applying good IT practices in accordance with cross-industry standards ensures an integrated approach to quality and information security management.

4.5.3.3 EU General Data Protection Regulation

Requirements for data processing security are stated in §32 (EU GDPR) [41]:

> *"The controller and the processor* [of data] *shall implement appropriate technical and organisational measures to ensure a level of security appropriate to the risk."*
>
> *"In assessing the appropriate level of security account shall be taken in particular of the risks that are presented by processing, in particular from accidental or unlawful destruction, loss, alteration, unauthorised disclosure of, or access to personal data transmitted, stored, or otherwise processed."*

Further, evaluation of security is a requirement described in Recital 83 (EU GDPR) [41], that

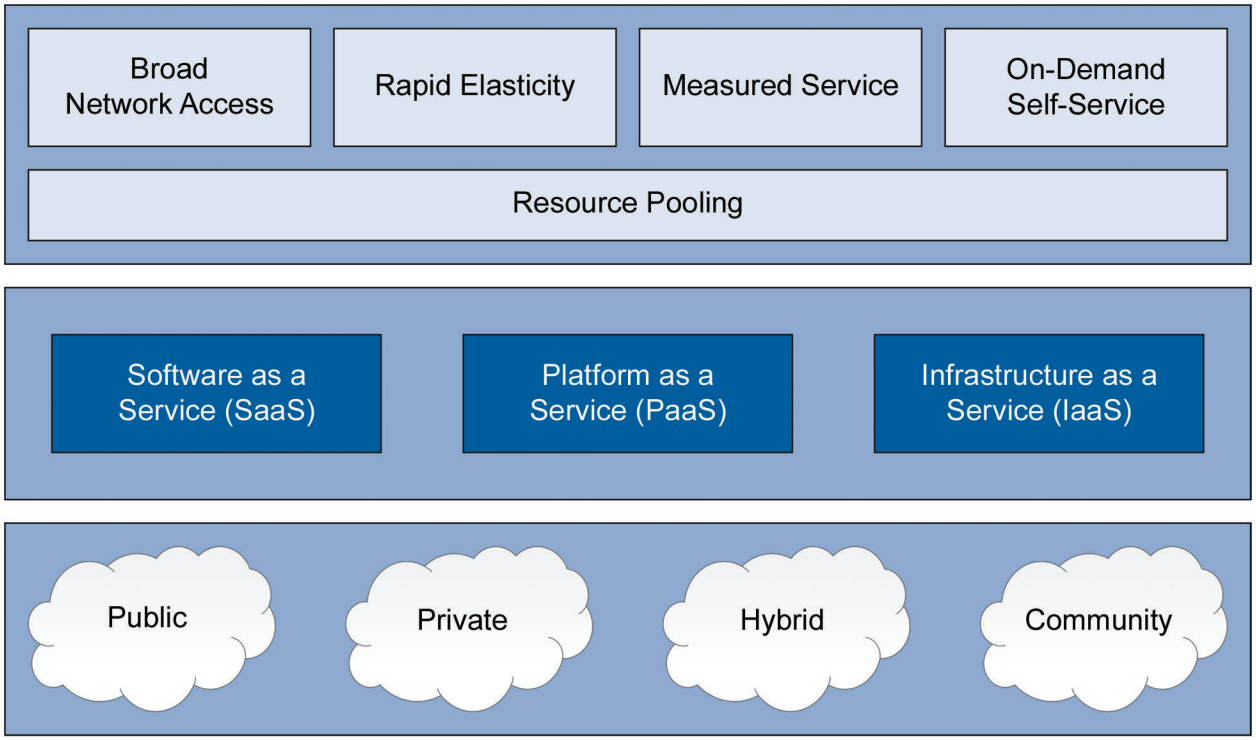
> *"the controller or processor should evaluate the risks inherent in the processing and implement measures to mitigate those risks, such as encryption. Those measures should ensure an appropriate level of security, including confidentiality, taking into account the state of the art and the costs of implementation in relation to the risks and the nature of the personal data to be protected. In assessing data security risk, consideration should be given to the risks that are presented by personal data processing, such as accidental or unlawful destruction, loss, alteration, unauthorised disclosure of, or access to, personal data transmitted, stored or otherwise processed which may in particular lead to physical, material or non-material damage."*

Adherence to EU GDPR [41] should provide additional assurance that general data protection controls are in place to support data integrity.

4.6 IT Service Models

There are a variety of different IT service models from traditional on-premises, outsourced, and cloud services such as IaaS, PaaS, SaaS, as shown in Figure 4.4.

Figure 4.4: Different Service Models [53]



The key components for an IT service model to be considered as cloud-based must include the following attributes:

- Availability from anywhere, notwithstanding it may be delivered as a private instance. Logically it can be accessible to all (if firewall rules were to allow).
- Self-service and on demand – can set up the service online and use it straight away
- Pooled resources – no individual service delivers the system so it should be able to grow and shrink within the limitations of a user's agreement with the provider
- Elasticity – the system or resource or service can grow, shrink, and move based on demand "automatically"
- Measured service – only pay for actual usage whether by metering or consumption, or by defined expectation

Figure 4.5: Scope of Services in Different IT Service Models [54]

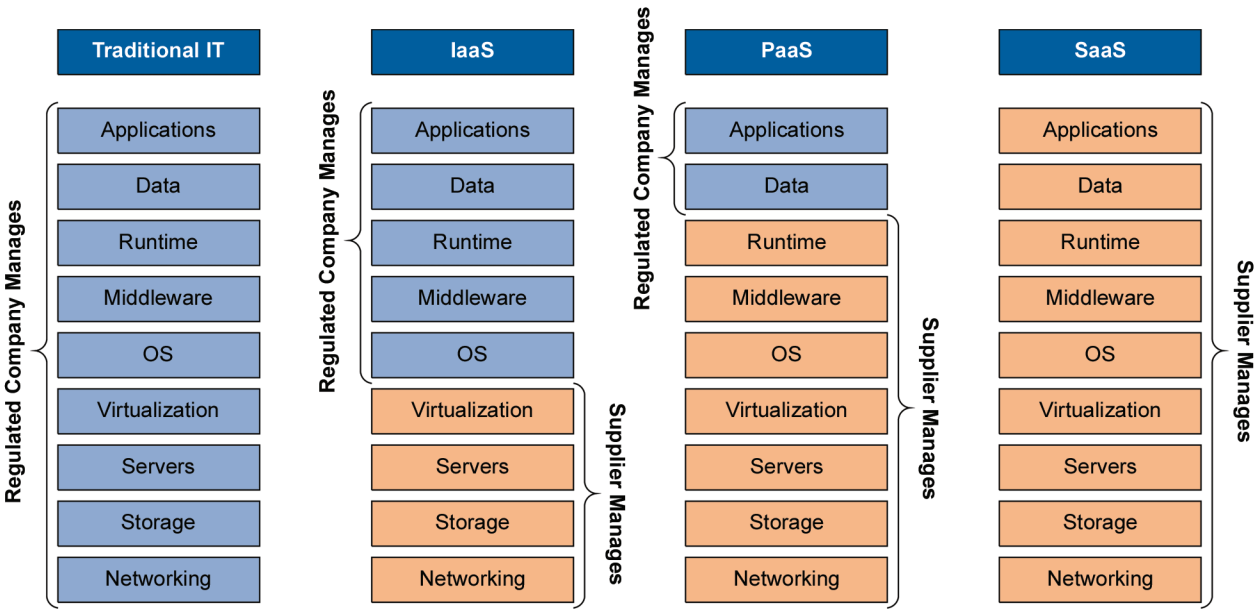


Figure 4.5 highlights the typical distribution of responsibilities between the regulated company and IT/IS service provider depending on the service model. The level of responsibility and dependency on the IT/IS service provider controls increases when providing SaaS services. Further, the complexity of the organizational relationship also increases as often the SaaS service provider engages the services of another organization to provide the infrastructure and platform services.

It should be noted that even with a full SaaS implementation, there is often a shared responsibility model between the service provider and the regulated company, for example, User Access Management often remains with the regulated company. In considering the distribution of roles and responsibilities for IT-related activities, it must be remembered that while IT-related activities may be delegated to IT/IS service providers, in all cases regulatory accountability lies with the regulated company.

XaaS computing takes many forms but only a number of these services could be denoted as utilizing true “as a Service” models. This section explains the different true service layers that are delivered and the approach that may need to be taken when approaching quality or validation of said environments.

Infrastructure as a Service (IaaS) is computing infrastructure, provisioned, and managed over the internet. IaaS is built to scale up and down with demand. This elasticity or consumption demand means the user only pays for what they use. Each IaaS resource is offered as a separate service component, and customer organizations only need to “rent” a particular one for as long as required. Cloud computing service providers manage the infrastructure as a “black box” that customers do not get access to, while these customers install, configure, and manage their own software: operating systems, databases, middleware, and applications.

Platform as a Service (PaaS) is a complete environment in the cloud, with resources that enable customers to deliver everything from simple cloud-based apps to sophisticated, cloud-enabled enterprise applications. These customers purchase the resources needed from a cloud service provider on a pay-as-you-go basis and access them over a secure internet connection. Like IaaS, PaaS includes infrastructure – servers, storage, and networking – but also middleware, development tools, Business Intelligence (BI) services, database management systems, and other applications. PaaS is designed to support the complete application life cycle: building, testing, deploying, managing, and updating.

Software as a Service (SaaS) allows users to connect to and use cloud-based apps over the internet. Applications include Customer Relationship Management (CRM), ERP, Clinical Data Management (CDM), electronic Trial Master File (eTMF), Track and Trace, as well as IT services such as productivity applications and email.

SaaS provides a complete software solution that is purchased on a pay-as-you-go basis from a cloud service provider. The regulated company essentially rents the use of a software application for their organization and only controls the configuration rather than install, configure, and test. Users generally connect to it over the internet or client software (that connects over the internet). The level of configuration permitted by the regulated company varies depending on the nature of the application. Some applications only permit the switching on and off of functional features, where others require more extensive configuration.

SaaS service providers may host the software application and data within their own data center or may engage the services of other IaaS and PaaS providers. With such models, it is essential that the roles of all organizations involved are defined, in particular the shared responsibilities for information security.

When looking at true cloud services from a validation or quality point of view, several new factors must be considered as all IT/IS service providers follow a similar pattern:

- Location – many services are only delivered by region, i.e., EMEA, US, APAC, etc. The exact physical location of a customer's specific virtual server will likely only be known to the IT/IS service provider. Customers may need to ensure that their system resides, or does not reside, in a particular country or region. This can be addressed during commercial negotiations and confirm that the same restrictions must be levied to any subcontractor or supplier used by the service provider to deliver the service.

- Hardware Configuration – a system specification is chosen at point of order. For example, if a customer orders a server with 4 CPU, 2TB standard drive space, 16GB Memory, and 1GB network, this is allocated as a virtual instance on a much bigger platform and there is no "qualification" of the environment except to confirm the delivered virtual instance matches the order. Automatically delivered build reports are often used to confirm the build matches the order. The process to "build" a server is measured in minutes.

- Operating Systems and Databases – like hardware configuration, customers can choose the version of operating system and database. Again this is deployed as the version chosen and is shown on the build report specified in the confirmation of the order.

- Patching – a risk-based approach should be taken to the deployment of patches. Security-related patches should be automatically deployed based on the frequency and criticality of security patch releases. Patches relating to application functionality and database platforms should be deployed in defined release windows with prior notification to the regulated company. For full multitenant, single instance SaaS, there is greater reliance on the SaaS provider's controls in managing patch releases.

- Elasticity – platforms can be moved live to larger environments as capacity is exceeded (or smaller if reduced), most of which will be seamless and unannounced to the customer. For high availability systems, the failover can happen instantly and may even be to another data center location. Disaster Recovery (DR) supports elasticity as well as high availability, and initiation of DR in case of a hardware outage can be automated. Testing is likely to be performed by the customer, but the responsibility lies with the service provider.

- Security monitoring and review – cloud tools are used to monitor security vulnerabilities and risks, including the review of perimeter security controls (e.g., firewalls).

A primary consideration of cloud services is that the regulated company does not own the data center, infrastructure, platform, or application (or a subset thereof) and must therefore assess, in accordance with risk, the adequacy of controls provided by the IT/IS service provider.

4.7 Risk Considerations

This section outlines some of the key risk considerations for adoption and use of the cloud IT service models within a regulated industry. These are not a definitive set of risks, and there are several frameworks available, most notably from the Cloud Security Alliance [38], ISACA [55], and NIST [37], which provide more comprehensive guidance on managing cloud risk.

To keep abreast of the most current key risks and vulnerabilities in the cloud, it is also recommended to consult resources such as the Cloud Security Alliance's "Top Threats to Cloud Computing" report [52], which is refreshed on an annual basis and aims to increase awareness over the key risks and vulnerabilities within the cloud computing landscape.

The primary risks resulting from a failure of an IT service, or the IT infrastructure environment, relate to [40]:

- Data protection, data integrity, and availability
- Business application availability and performance

Risk assessments must take account of the primary threats to cloud services such as:

- Infrastructure component/environment failures
- Security breaches including cybersecurity
- Software failures
- IT/IS service provider failures

4.7.1 IT Infrastructure Risks

IT infrastructure architectures (on-premises and in the cloud) employ widely used industry-standard components (GAMP Category 1 hardware and software [2]) that typically include error detection and self-correction features, leading to a low failure rate and a high probability of threat detection.

> *"These risks are continuous, and it is therefore imperative that the currency of IT infrastructure controls is maintained (e.g., through patching) and that monitoring is in place to provide early detection of any threat. IT service and infrastructure design incorporates a high degree of resilience that mitigates both single-point and complete failure."*

> *"Further, IT infrastructure supports business applications that hold, process, and transmit regulated records. The completeness and accuracy of these regulated records are largely governed by the business processes supported by these business applications. IT infrastructure must provide a secure platform that hosts these applications and data but does not directly impact regulated records. This ensures that the risk to patient safety, product quality, and data integrity resulting from an IT service or infrastructure failure is low."* [40]

IT services and infrastructure also support nonregulated business applications and data. Therefore, the IT infrastructure cannot effectively be partitioned into GxP and non-GxP. Instead, common IT practices and controls are used to manage IT infrastructure supporting both GxP and non-GxP operations. [40]

> *"Industry-standard IT management practices, electronic service management tools, modern IT service models, automation, and continuous monitoring are essential for ensuring the performance, security, and integrity of the IT infrastructure environment."* [40]

4.7.2 Information Security Risks

Information security risk can be associated with unauthorized alteration, loss, leakage, or unavailability of data. This can cause data breaches and business interruption as well as regulatory compliance risks to regulated customers.

In the Cloud Security Alliance's 2019 "Top Threats to Cloud Computing" report [52], *data breach* is ranked as the number one cloud threat. Data is becoming the primary target for cyberattacks, and data accessible via the internet is the most vulnerable asset to misconfiguration or exploitation.

It is therefore vital that the regulated company fully understands the value of the data handled in the cloud, and the impact of its loss, leakage, or unavailability. Data classification is a vital step toward this and to building an effective cloud security control environment. Information owners should be engaged to assess and classify information assets based on business risk. This reduces unnecessary security expenditure as more resources are focused on protecting the most critical GxP data. This should also help inform the acceptable risk tolerance for moving data to the cloud and adopting an appropriate service model.

When a regulated company moves data into the cloud, consideration must be given to who will have access to this data. The key concern is how access to GxP (and privacy-related) data by the IT/IS service provider or their service partners is controlled to ensure that even Administrator access is based on least privilege and segregation of duties.

4.7.3 Software Failures

There is a greater dependency on the IT/IS service provider's QMS when using SaaS. In particular, for single instance, multitenant services, there is a greater need to leverage the IT/IS service provider's effort in support of the overall validation effort. (See Section 2.3.5.)

Adherence to a robust IT QMS is essential in the development and verification of SaaS applications. Further, IT/IS service provider activities, documentation, and/or records must provide demonstrable evidence to the regulated company that the service is fit for intended use.

These activities, documentation, and records are evaluated during the regulated company's assessment of the IT/IS service provider and summarized in the regulated company's reports and can be used to demonstrate to regulators that appropriate controls are in place.

4.7.4 IT/IS Service Provider Risk

The fast growth within the SaaS industry has also resulted in an increasing number of start-up SaaS companies. Although these companies may offer attractive IT service and software solutions, there is a risk that they may not operate adequate quality and information security management systems. There is also a risk over the long-term viability of such organizations.

In particular, with SaaS, ownership and control of the IT infrastructure and software application is with the IT/IS service provider. A business failure of such a service provider could severely impact business continuity for the regulated company.

Furthermore, failure to operate appropriate quality and information security management systems significantly impacts the ability of the regulated user company to leverage the IT/IS service provider efforts in support of their own regulatory obligations.

A regulated company must be able to demonstrate effective due diligence over IT/IS service provider selection and ongoing IT/IS service provider management. Appropriate policies and controls should be in place to prevent cloud services from being procured directly through "non-approved" routes, such as business or shadow IT groups. A key consideration in the selection of a cloud service provider (IT/IS service provider) is to ensure that the IT/IS service provider has adequate controls in place (and is able to demonstrate such) to comply with various laws, rules, and regulations. This includes data privacy aspects, as IT/IS service providers may be based in countries that do not always meet the expectations of the EU GDPR [41], for example. Data ownership should also be fully understood, along with the potential risk that data could be used for secondary purposes (such as the SaaS provider using anonymized data to train Artificial Intelligence (AI) models). Contractual provisions should define the permitted access to the regulated company's data, that is, in connection with the provided services.

4.7.5 Data Ownership and Portability

The regulated company always owns the data managed by the cloud services (XaaS) and must ensure that controls are in place to minimize the risk to data integrity, data loss, and data availability.

Data portability may become a risk, especially when SaaS platforms become embedded within the regulated company's ecosystem and there are upstream and downstream integrations. If IT/IS service providers do not use standard Application Programming Interfaces (APIs), protocols, and tools, it may be more difficult and/or expensive to move to another IT/IS service provider. Organizations should formulate exit strategies or contingency plans to extract or migrate their data (including metadata) to an alternative solution in the event of contract termination.

Contracts and/or quality agreements need to define expectations for retrieval of data and associated metadata in a readable (and where necessary processible) form to facilitate ongoing access and/or migration to another system or format. See Section 4.9.3.

4.8 IT/IS Service Provider Governance

The regulated company is accountable for regulatory compliance of IT infrastructure and software solutions. The IT/IS service provider follows their internal QMS, which governs the delivery of IT/IS services in a controlled manner. Regulatory accountability does not change with the use of cloud services. The regulated company needs to determine the risks to the patient safety, product quality, and data integrity when outsourcing services to a third party. Consideration should also be given to any sub-IT/IS service providers supporting the overall services provided, and the quality management of those lower-level services, such as IaaS, should be accounted for in the overall service delivery model.

The approach to IT/IS service provider assessment and governance should be based on a documented risk assessment of the scope and impact of the services being delivered with respect to the regulated business processes and data. The quality assessment is a subset of the overall assessment of the IT/IS service provider capabilities, which includes assessment by IT, business, information security, data privacy, etc. These assessments should be coordinated by the regulated company to minimize duplication across different assessments and to ensure consistency of approach.

To determine the level of IT/IS service provider governance required, the regulated company must understand the risk to patient safety, product quality, and data integrity presented by the services being procured.

As indicated in Section 4.4, the IT/IS service provider must have the appropriate policies and procedures in place to provide adequate quality and information security management.

Similarly, the regulated company must have the appropriate processes in place to assess the controls applied by the IT/IS service provider and to leverage the IT/IS service provider's effort.

Governance controls should also ensure that MSA, SLA, and/or quality agreements adequately define the quality controls and quality metrics required of the IT/IS service provider.

The level of oversight and rigor should be dependent upon the intended use employing a risk-based approach as described in *ISPE GAMP 5,* Section 5 and Appendix M3 [2], in alignment with ICH Q9 [4] and ISO 14971 [56].

4.9 IT/IS Service Provider Assessment

The approach to the IT/IS service provider assessment should be based on a documented risk assessment of the services provided as discussed in Section 4.7. Based on risk, assessment of IT/IS service providers should leverage readily available IT/IS service provider records such as independent attestations, internal assessments, and certifications as well as records accessed through the relevant assessment process. Combined with the Committee of Sponsoring Organizations of the Treadway Commission (COSO) [57] principles and trust services alignment, they provide an approach to assessing if the IT/IS service provider aligns with IT practices.

IT infrastructure and platform services can be assessed through the evaluation of a range of certifications and attestations made available by the IT/IS service provider. These include SOC 2+ reports (see July/ August 2019 *Pharmaceutical Engineering* article "Application of the SOC 2+ Process to Assessment of GxP Suppliers of IT Services" [58]), ISO 9001 [35], ISO 27001 [36], ISO 27002 [59], ISO 27017 [60], ISO/IEC 20000-1 [61], and other certifications. NIST Special Publication 500-322 [62] provides information on each service type. Evaluation of these materials helps ensure quality and compliance with quality obligations and IT security.

The assessment approach for SaaS providers should be based on a documented risk assessment. Additional rigor should be applied where there is significant risk to patient safety, product quality, or data integrity, and dependence on service provider quality systems (e.g., multitenant, single instance solutions). The risk assessment should also determine the need for ongoing IT/IS service provider assessments throughout the period of IT/IS service provider engagement.

The requirement to periodically reassess the IT/IS service provider and the frequency of such assessments (including for-cause assessments) should be stated in the MSA and/or quality agreement.

IT/IS service providers may utilize third-party organizations in the provision of services and solutions. It is common for SaaS providers to use service providers such as IaaS to host their applications or software development and support organizations. Based on a documented risk assessment, the regulated company should determine adequacy of the IT/IS service provider's subcontractor governance processes.

It may be helpful to leverage specific cloud control frameworks (e.g., Cloud Security Alliance (Consensus Assessment Initiative Questionnaire (CAIQ)) [63]) to create a tailored set of IT/IS service provider assessment queries that reflect the different types of controls applicable to the IT/IS service provider.

4.9.1 Multitenancy

In multitenancy, computing capacity, storage, and network are shared across multiple cloud customers. While this model allows cloud providers to achieve economies of scale and lower service costs, there is increased risk that a single vulnerability or misconfiguration can lead to a data compromise across multiple customers. Multitenancy in cloud service models implies the need for policy-driven enforcement, segmentation, isolation, governance, and service levels for different cloud customer groups.

For regulated companies, the key risk consideration is to ensure that data is effectively segregated from other customers. The regulated company should ensure that appropriate controls are in place to segregate their data from other organizations, for instance logically within the application or at the database layer. The IT/IS service provider should be able to provide details of the technical controls implemented to ensure appropriate segregation. This emphasizes the need to fully understand the underlying architecture that underpins the services the regulated company has contracted.

4.9.2 Evidential Sources

Many IT/IS service providers can provide evidence that demonstrates appropriate controls are in place, from formal quality systems through evidence of how they ensure the confidentiality, integrity, and availability of systems. The level of assurance offered can vary in scale from client-facing statements of systems of controls, IT/IS service provider self-assessment of controls, to independent assessment and attestation of controls and formal certifications.

The level of reliance that can be placed on this evidence depends upon the scope of the controls assessed and the level of independent scrutiny obtained. Mapping control expectations to actual controls and evidence sources helps the regulated company focus effort on areas that are either missing from the IT/IS service provider assessments or have deficiencies identified in the design or operating effectiveness of controls.

Evidence over the system of controls can come from a combination of sources. Common examples include:

4.9.2.1 Cloud Security Alliance Security Trust Assurance and Risk Program

The Cloud Security Alliance's Security Trust Assurance and Risk (STAR) program [63] allows IT/IS service providers to validate and offer proof of their security controls. There are multiple levels of STAR assurance from self-certification to third-party attestation and/or certification. The IT/IS service provider needs to be registered with the STAR program and agree to share reporting with existing/prospective customers.

4.9.2.2 System and Organization Controls

System and Organization Controls (SOC) evaluations are produced by Certified Public Accountants (CPAs) and follow guidance from the American Institute of CPAs (AICPA) [64]. There are three types of SOC reports on the internal controls of service organizations: SOC 1®, SOC 2®, and SOC 3®.

Table 4.2: SOC Report Contents [64]

Report	SOC 1® – SOC for Service Organizations: ICFR	SOC 2® – SOC for Service Organizations: Trust Services Criteria	SOC 3® – SOC for Service Organizations: Trust Services Criteria for General Use Report
Description	Report on Controls at a Service Organization Relevant to User Entities' Internal Control over Financial Reporting (ICFR) Type 1 (attestation of controls at a specific point in time) Type 2 (attestation of controls over a minimum period of 6 months)	Report on Controls at a Service Organization Relevant to Security, Availability, Processing Integrity, Confidentiality, or Privacy Type 1 (attestation of controls at a specific point in time) Type 2 (attestation of controls over a minimum period of 6 months)	General use Report on Controls at a Service Organization Relevant to Security, Availability, Processing Integrity, Confidentiality, or Privacy
Availability	Limited distribution	Limited distribution	General availability
Relevance	Financial reporting only	Type 2 report is preferable	Too high level

The SOC 2 report is likely to be the most relevant. Based upon the 2017 Trust Services Criteria [65] (established by the AICPA's Assurance Services Executive Committee), it provides detailed information concerning the security, availability, and processing integrity of a service provider's processing environment. These reports are available as two versions Type 1 or Type 2. Type 1 reports provide a description of the providers' system and the suitability of the design of controls. Type 2 reports expand upon Type 1 by also assessing the operating effectiveness of controls over a period of time, with control testing performed by the service auditor.

Enhanced SOC 2 reports, also known as SOC 2+ reports, can be used to demonstrate assurance in areas that go beyond the Trust Services Principles, such compliance with regulatory and industry frameworks. An approach to assessing IT/IS service providers is proposed in "Application of SOC 2+ Process to Assessment of GxP Suppliers of Services" [58].

4.9.2.3 Evaluating the SOC Report

When evaluating SOC reports, focus on:

- The scope of the assessment: what processes and controls are covered
- The time period covered
- The outcome of control testing: any testing exceptions should be evaluated for their significance and followed up if needed

Where subcontracted service providers are used by IT/IS service providers, they will be outside the scope of this specific report, but IT/IS service providers may be able to provide equivalent information for any subcontracted services. It is also possible for the SOC report to include additional information provided by the service organization not covered by the service auditor's report. While this may provide useful background information, it is not subject to independent verification, so it should be treated with caution.

Information Security Certifications – ISO/IEC 27001 [36], ISO/IEC 27017 [60]: IT/IS service providers should be able to demonstrate sound information security practices if they are compliant with ISO/IEC 27001. In addition, IT/IS service providers should also comply with ISO/IEC 27017, which is used as an extension to ISO/IEC 27001 and provides enhanced controls for both IT/IS service providers and cloud service customers. In addition to obtaining a current certification, a statement of applicability should be supplied, which defines the specific scope of the certification. IT/IS service providers may also be willing to share their most recent ISO 27001/27017 audit reports.

Quality System – ISO 9001 [35]: IT/IS service providers should be able to demonstrate that a formal QMS is established and followed. The IT/IS service provider should either have a formal ISO 9001 certification, or a QMS that is demonstrably equivalent. It is important to ensure that the QMS is applicable for the specific contracted services and locations.

Evaluation of SOC reports should be conducted on a periodic basis in accordance with risk in order to ensure the state of control is continuously maintained.

4.9.2.4 Supply Chain

There are often subcontractor relationships in place, for example, a SaaS provider may engage with an IaaS/PaaS provider. The regulated company's assessment strategy, based on a documented risk assessment, should ensure that all IT/IS service providers are appropriately assessed based on the potential impact on the supported regulated process and data. This is largely achieved by ensuring the primary contractor has adequate controls in place to assess their subcontractors.

4.9.2.5 Evidential Considerations

Evidence provided by modern IT/IS service providers to demonstrate that appropriate controls are in place typically differ from historical, traditional, IT qualification documentation. As discussed in Section 4.4, electronic systems, service-management tools, and automation may underpin the IT QMS. It is therefore essential that the IT/IS service provider assessments evaluate evidence against the IT controls objectives, rather than preconceived expectations of approach or documentation/records set.

4.9.3 Master Service Agreements, Master Subscription Agreements, Quality Agreements, and Service Level Agreements

MSAs, master subscription agreements, quality agreements, and SLAs collectively define the quality expectations of the regulated company. These contractual documents may also be combined, where the quality agreement and SLA form appendices to the MSA. These documents should define the quality objectives to be met by the IT/IS service provider and the service level metrics to be achieved to demonstrate that the required service levels are being met. The regulated company should take care not to impose their internal ways of working on the IT/IS service provider, as the service provider is working for multiple regulated companies with different ways of working.

IT/IS service providers may also have generic MSAs, quality agreements and/or SLAs that may be reviewed and adopted by the regulated company where acceptable. Such generic agreements often offer tiered service levels depending on the nature of the services being provided.

The roles and responsibilities section of the quality agreement should delineate the responsible party as either the regulated company, IT/IS service provider, or both for a given quality requirement. For each responsibility listed, the regulated company is establishing the controls to support the quality obligations of their GxP requirements. The IT/IS service provider is agreeing to fulfill these requirements by utilizing controls to support the regulated company's quality obligations. The IT/IS service provider is responsible for assuring and managing its subcontractors, and needs to be able to demonstrate the appropriate controls.

The level of control defined within these contracts and agreements is influenced by the criticality of the provided services as defined in the service requirements specification.

For comparison purposes, Table 4.3 summarizes the typical contents of MSAs, SLAs, and quality agreements.

Table 4.3: Typical Contents of MSAs, SLAs, Quality Agreements, and Data Protection Agreements

Document Type	Typical Contents
Master Service Agreement/ Master Subscription Agreement	• Definition of Services • Protection of Data • Responsibilities of Contracted Parties • Permitted Service Locations • Deliverables • Term of Contract • Intellectual Property Rights • Warranties • Dispute Resolution • Legal Terms and Conditions • Payment Terms • Force Majeure • Confidentiality • Contract Termination • Insurance Requirements

Table 4.3: Typical Contents of MSAs, SLAs, Quality Agreements, and Data Protection Agreements (continued)

Document Type	Typical Contents
Service Level Agreement	• Definition of Services • Service Performance Metrics (e.g., minimum uptime, maximum downtime) • Upgrade Management • Patch Management • Release Cycles • Support Hours • Support Levels and Response Times • Performance Metrics (Response Times, Service Availability, etc.) • Issues Reporting and Management
Quality Agreement	• Definition of Services • Quality Responsibilities of Contracted Parties • Requirements of the IT/IS Service Provider Quality Management System • Interfaces between Customer and IT/IS Service Provider Quality Systems (where appropriate) • Quality Issue Escalation • Reporting Requirements • Audit Requirements
Data Protection Agreement (Data Privacy)	• Data Processing and Sub-Processing Controls • Data Transfer Controls • Personnel Controls • Third-Party Controls • Information Security • Management and Reporting of Data Breach

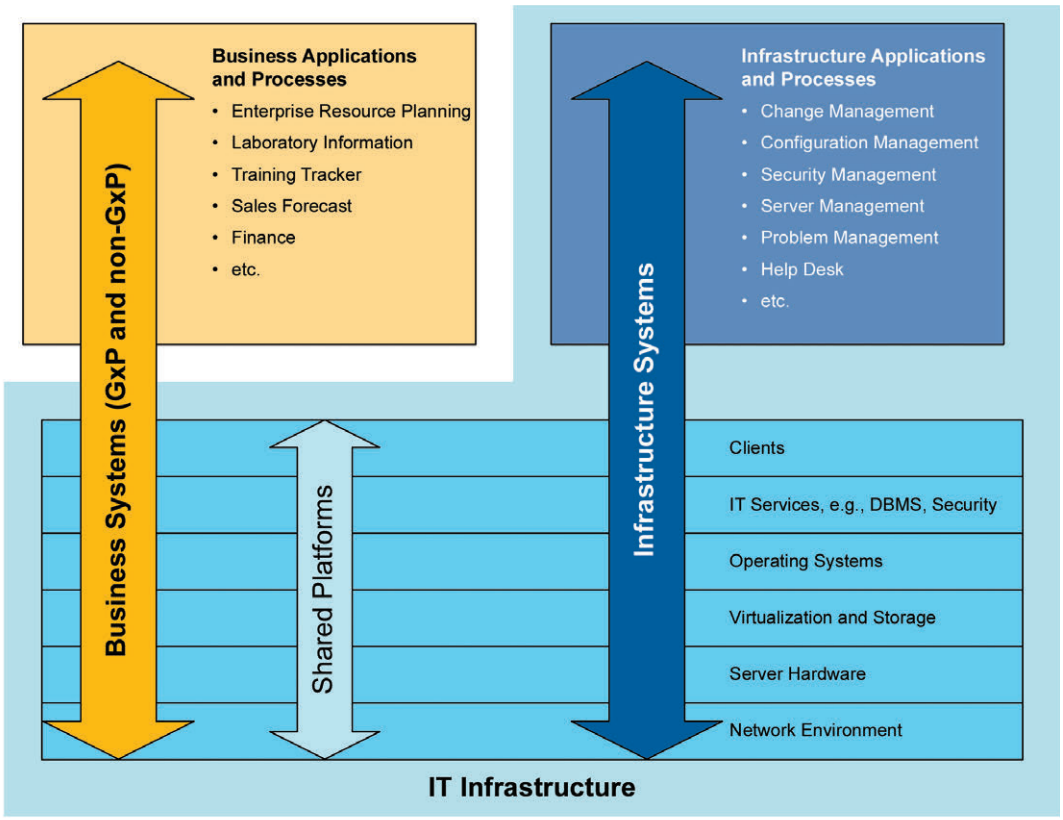
4.10 Management of IT Infrastructure

Historical IT qualification processes based on paper records are typically inefficient. Such approaches often only confirm the status of the IT infrastructure at a point in time and seldom ensure the correctness of the ongoing operation, availability, performance, and protection of the IT environment. Such traditional qualification processes seldom provide assurance that IT controls continue to operate effectively and are not effective in managing threats and risks.

> *"Advances in IT service models, virtual technologies, automation, monitoring, and self-correcting technologies have led to significant improvements in IT governance. Implementation of an IT quality management system based on industry standards, electronic service management tools, and automation is fundamental to managing IT risks."* [40]

The *ISPE GAMP® Good Practice Guide: IT Infrastructure Control and Compliance* [54] defines typical IT infrastructure components and processes that form the IT QMS (Figure 4.6).

Figure 4.6: IT Infrastructure Model [54]



The IT infrastructure provides a controlled environment within which business applications operate in support of regulated business processes. Traditionally, IT qualification practices have been employed to ensure that the IT infrastructure is appropriately specified, designed, configured, and deployed. Advances in IT practices, service models, service management tools, and automation provide an opportunity to establish and maintain the qualified status of IT infrastructure in a robust and efficient manner that minimizes the risk of IT service and infrastructure failures.

4.10.1 Embracing Infrastructure Automation

"As reported in a panel discussion, "FDA and Industry Collaboration on Computer Software Assurance (CSA)" at the Institute of Validation Technology's 20th annual Computer and IT Systems Validation conference, 23 April 2019, the FDA and industry team's recommendations are to:

- *Embrace automation in the management of IT infrastructure.*
- *Use electronic means rather than paper documentation.*
- *Leverage continuous data and information for monitoring and assurance."* [40]

This approach improves quality and process control while lowering quality, security, and integrity risks.

The industry team reported case studies on replacing manual, paper-based, and error-prone test evidence and specification maintenance with an automated, error-free approach based on standard tools. In these case studies, the time savings were 10-fold (i.e., the automated approach takes only a tenth of the time of the manual method). [40]

4.10.2 Service Management Tools

> *"Electronic service management tools that incorporate configuration management databases (CMDB) and electronic workflows supporting change management, configuration management, and incident and problem management are integral to the IT quality management system."* [40]

The CMDB supports effective management of the configuration status of IT services and IT infrastructure components and business applications. Electronic workflows ensure adherence to processes and collaboration of IT SMEs across global organizations. [40]

4.10.3 Integrating Traditional IT Qualification Controls

Traditional qualification activities can be integrated into the IT QMS and service management tools, avoiding the need for one-off protocols and paper-records management. For example, for a backup service, work instructions can be created within the service management tool to define how a new server or storage device is added to the backup solution. Backup scheduling is configured within the backup tool to ensure backups are scheduled at the right frequency. Alerts are configured to automatically notify of failures. [40]

Automated feedback of successful backup completion is provided in support of the periodic backup testing. Evidence supporting these automated operations (e.g., backup status logs) can be retained as evidence of continuous backup verification.

This can be achieved without the need to create and execute stand-alone protocols. In essence, the backup deployment, configuration, and monitoring become part of the operational processes of the IT QMS. [40]

4.10.4 IT Services and Infrastructure Monitoring

Monitoring technologies provide real-time feedback on the status of the IT services and IT infrastructure. Monitoring includes, but is not limited to:

- Information security vulnerabilities
- IT environment availability
- Database performance
- IT component failure
- Network connectivity issues
- Application and platform errors
- Virtual environment performance

> *"Machine learning is now being deployed to evaluate data sets (events and logs) generated by monitoring tools. Data trends are analyzed to predict potential IT incidents and proactively act to minimize the risk of failures."* [40]

4.10.5 IT Life Cycle and Automation, Utilizing IT Monitoring Tools

IT infrastructure service and device management comprises:

- IT resource provisioning
- Configuration management

- Monitoring
- Configuration auditing
- Service optimization

"Resource provisioning utilizes Infrastructure as Code (IaC) and virtual machine templates to provision new servers and services that are configured in accordance with IT standards. Infrastructure code and templates are subject to version control using code management tools. Changes to code and templates are fully auditable in the event of an inadvertent or unauthorized change." [40]

Configuration management ensures that infrastructure code and templates automatically provision standard configurations. Code and templates can be verified once and used many times when provisioning like resources.

Monitoring uses tools that monitor IT availability, performance, incidents, and security vulnerabilities. Automated alerts are directly sent to support teams to enable a timely response. Self-correcting technologies allow for adjustments in configuration to address reported issues. Security log monitoring identifies and reports potential unauthorized access attempts. [40]

Configuration auditing monitors deviations from standard configurations. Environments are automatically audited against standard configurations, and deviations can be self-corrected following inadvertent or unauthorized change.

"Optimization is enabled through metrics provided by monitoring tools. IT resources such as processing capacity, storage capacity, database capacity, network routing, and load balancing can be adjusted based on feedback to maintain system availability and performance." [40]

4.11 Validation of SaaS Applications, Demonstrating Fitness for Purpose

This section provides guidance for applying *ISPE GAMP 5* [2] to the validation of SaaS solutions.

ISPE GAMP 5 [2] provides extensive guidance on how to achieve and maintain a compliant GxP computerized system that is equally applicable to the validation of GxP computerized systems based on SaaS solutions. The key concepts of *ISPE GAMP 5* that are of particular importance are understanding intended use, applying risk management, and leveraging IT/IS service provider effort, which, if applied correctly, can minimize the software validation activities that the regulated company has to perform.

The key concepts should be underpinned by the implementation of software change management at the start of the software validation process. In addition, there needs to be a recognition that the IT/IS service provider of the SaaS solution is not subject to GxP regulations. This means that leveraging IT/IS service provider input should shift from a life cycle *deliverables*-based approach to a life cycle *effectiveness*-based approach.

With a *life cycle deliverables-based* approach, specification and verification deliverables are produced by the IT/IS service provider and the regulated company expects to receive a copy of, or access to, the IT/IS service provider's deliverables (e.g., validation package or installation/operational qualification package). In addition, there is often an expectation that deliverables are signed documents.

With a *life cycle effectiveness-based* approach, the focus shifts to ensuring the ongoing effectiveness of the IT/IS service provider's activities and an acceptance that the IT/IS service provider's evidence typically will not be in the form of signed documents, but in structured, well-controlled and managed information in various forms within the service provider QMS.

ISPE GAMP 5 [2] identifies three applicable software product categories:

- Category 3: Non-Configured Products
- Category 4: Configured Products
- Category 5: Custom Applications

SaaS solutions typically have the characteristics of Category 4: Configured Products. For simplicity, this section assumes that SaaS solutions are Category 4 and are able to be preconfigured by the IT/IS service provider and then configured by the regulated company.

This section does not repeat the *ISPE GAMP 5* [2] guidance but instead focuses on the specific validation challenges presented by the adoption of SaaS solutions, and presents a recommended validation strategy to overcome these challenges. The recommended validation strategy is guidance only and should be revised to meet the regulated company's specific requirements as appropriate. Further details are given in Sections 4.10.1 and 4.10.2.

The transition to SaaS and cloud services in general requires a greater understanding of current practices for delivery and maintenance services through the use of Agile and DevOps methods and automation (see Chapter 3). Without such understanding, it will be difficult to assess the adequacy of IT/IS QMS and controls.

In particular, for SaaS solutions that require regulated company-specific configuration, a prerelease environment is necessary to enable the regulated company to conduct validation activities against their business process-specific configuration prior to release into a production environment. The IT/IS service provider should provide an adequate window for these activities to be completed.

4.11.1 Computerized System Validation Challenges

The adoption of SaaS solutions brings specific challenges to regulated companies, such as:

- The SaaS solution infrastructure is not controlled by the regulated company
- The installation or deployment of a SaaS solution on to the infrastructure is not controlled by the regulated company
- Dependency on OSS libraries
- The SaaS solution is supported by operational processes, many of which are not controlled by the regulated company, e.g., DR
- Access to a regulated company's data held by the SaaS solution is not fully controlled by the regulated company
- The risk of cybersecurity threats may be increased depending on the IT/IS service provider's maturity. With appropriate and mature service providers, the cybersecurity risk is likely to be lower
- There can be an accelerated deployment of software changes, i.e., enhancements and defect fixes, and the accelerated deployment schedule is not controlled by the regulated company. Often, these software changes are automatically enabled, and the regulated company is given limited time to assess and, if appropriate, verify these software changes. Furthermore, if the regulated company identifies a defect and the IT/IS service provider chooses not to roll back the software change, then the regulated company has no ability to reject the software change. Software changes may also occur when the initial validation is in progress.

- The IT/IS service provider may not produce evidence to demonstrate specification and verification activities in the "traditional" form that regulated companies have become used to, e.g., signed documents, and any evidence the IT/IS service provider produces may not be readily available to, or easily consumable by, the regulated company, e.g., test evidence and traceability data residing in a modern automated testing tool.

4.11.2 Recommended Validation Strategy

ISPE GAMP 5 [2] defines computerized system validation as:

> *"Achieving and maintaining compliance with applicable GxP regulations and fitness for intended use by:*
>
> - *the adoption of principles, approaches, and life cycle activities within the framework of validation plans and reports*
>
> - *the application of appropriate operational controls throughout the life of the system"*

The objective of computerized system validation does not change for a SaaS solution. It is still the regulated company and not the IT/IS service provider that must be able to demonstrate, with evidence, that compliance to applicable GxP regulations and fitness for intended use has been achieved and is maintained. The regulated company should establish a computerized system validation plan, or similar, to define a validation strategy that meets the objective, while addressing the challenges specified in Section 4.11.1. The challenges relating to infrastructure qualification, operational processes, access to data, and cybersecurity threats are addressed in Section 4.5. This section (4.11.2) focuses on ensuring that systems are fit for intended use.

From the list of challenges specified in Section 4.11.1, it is the regulated company's lack of control of software changes that causes the most difficulties. Assurance processes must ensure that the IT/IS service provider's QMS has effective controls governing the release of software changes.

The software validation strategy should aim to minimize the verification and assurance activities performed, and the evidence produced, by the regulated company. This is realized by eliminating any activity or evidence that does not add value to the objective of achieving and maintaining compliance with applicable GxP regulations and fitness for intended use. Specific areas to consider include ensuring that:

- Non GxP-intended use (i.e., intended use that does not impact patient safety, product quality, or data integrity) is not included in the software validation scope

- The focus is on functional requirements only. Nonfunctional requirements (e.g., backup and restore) are important but should be treated as operational processes and included elsewhere in the non-software parts of the life cycle strategy for the GxP computerized system.

- The focus of testing is on finding defects that impact the GxP-intended use and not on generating *non-value-added* evidence. The need to produce scripted test cases and supporting documentation, such as screenshots, should be carefully considered and eliminated if possible.

- *Non-value-added* reviewers, approvers, and signatures are eliminated

- Evidence produced is necessary to demonstrate compliance with applicable GxP regulations and fitness for intended use rather than for the benefit of internal auditors or regulatory inspectors.

A recommended software validation strategy that shows the regulated company's activities and evidence is presented in Table 4.4. It is not intended to be prescriptive but rather to give an example of the activities that the regulated company may perform. It is guidance only and should be revised to meet the regulated company's specific requirements as appropriate.

Prior to Activity 1 in Table 4.4, it is assumed that:

- A supplier assessment of the IT/IS service provider has been carried out. When doing the assessment, the regulated company should recognize that the IT/IS service provider is not subject to GxP regulations. However, they are responsible for ensuring that they have integrated quality controls and industry best practices into their development and operational processes, which means that the assessment should focus on assessing the effectiveness of these processes.

- With one exception, the IT/IS service provider's responsibilities have been agreed and included in a contract or quality agreement. Specifically, responsibilities regarding the software release management process should be agreed, such as the release schedule, release types and their content (e.g., the release of automatically enabled/disabled enhancements/defect fixes in major, patch, and hot-fix releases), the time window allowed and environment available for UAT, and release material published.

Note: If the IT/IS service provider is not able to provide the regulated company with a time window or make available an environment for UAT, then the regulated company should give serious consideration as to whether the SaaS solution is suitable for GxP-intended use. The quality agreement should not delegate software validation responsibilities to the IT/IS service provider.

The exception is that as part of the risk assessment (see Activity 6 in Table 4.4) the regulated company might want the IT/IS service provider to perform additional activities as a risk-reduction control measure for specific high-risk functionality. If the IT/IS service provider agrees to do this, then the contract or quality agreement should be revised as appropriate.

Prior to Activity 8 in Table 4.4, it is assumed that the regulated company has a business continuity process established that can be invoked if a critical defect occurs that is not fixed by the IT/IS service provider in an acceptable time frame.

Table 4.4 Recommended Validation Strategy

Note: The validation strategies defined here are intended to ensure that outsourcing of IT/IS services to third-party organizations does not lead to an increased risk to patient safety, product quality, or data integrity.

Id.	Activity	Evidence	Further Details
1	Define the validation strategy	Validation Plan	The Validation plan specifies the activities and evidence given below.
2	Define and initiate the software change management process **Note:** The regulated company should consider creating an initial Change Management Record for the software release, which is current at the start of validation.	Software Change Management SOP Change Management Record for the software release current at the start of validation	Because the regulated company's system validation process and the IT/IS service provider's System Development Life Cycle are running in parallel and are not necessarily synchronized, achieving and maintaining compliance with GxP regulations and fitness for intended use should be addressed by a software change management process, which is initiated at the start of the validation process. The regulated company cannot avoid assessing the impact of every relevant and significant software change on their intended use. However, they can be creative in how they record the assessment, e.g., only creating a Change Management Record for software changes that have an impact, logically grouping software changes to minimize the number of change management records, etc. The software change management process should also be able to address the scenario when an automatically disabled software change is not enabled initially by the regulated company but is enabled at a later date. The software change management process should specify a regulated company's activities and evidence only. Any evidence produced by the IT/IS service provider is not used as part of the regulated company's "validation package." For example, if the IT/IS service provider is able and agrees to provide a software development deliverable, then the software validation activity is the review by the regulated company of the IT/IS service provider's software development deliverable, and the software validation evidence is the record that the IT/IS service provider's software development deliverable was reviewed. All remaining validation activities should be carried out in accordance with the Software Change Management SOP.
3	Understand software functionality delivered by SaaS solution	N/A	The regulated company should ensure that they understand the functionality delivered by the SaaS solution. There are multiple ways in which an understanding of the system functionality can be obtained, e.g., reviewing specifications produced by the IT/IS service provider if they exist and are accessible, reviewing user documentation or online help, hands-on access to a sandbox environment, etc.

Table 4.4 Recommended Validation Strategy (continued)

Id.	Activity	Evidence	Further Details
4	Specify the GxP-intended use **Note:** If software changes occur while activity is in progress, then Activities 3 and 4 are repeated. Changes are tracked in Change Management Record(s).	User Requirements/ Configuration Specification Change Management Record(s) if required	The regulated company should assess how they intend to configure and use the SaaS solution to support business processes that have a GxP impact, i.e., impact on patient safety, product quality, and data integrity. As part of this assessment, the regulated company should consider whether the GxP impact can be eliminated by a process redesign, e.g., including robust and reliable in-process or downstream detectability controls. The output from the assessment should be a User Requirements Specification, against which a risk assessment can be conducted that the potential impact of functional failures can be assessed against patient safety, product quality, and regulated-data integrity. In addition, as part of the User Requirements Specification or in a separate Configuration Specification, any preconfiguration by the IT/IS service provider or configuration by the regulated company should be specified.
5	Assess the risk of software functional failure and identify risk-reduction controls that can be implemented or performed by the regulated company **Note:** If software changes occur while activity is in progress, then Activities 3 to 5 are repeated. If GxP-intended use is revised while activity is in progress, then Activities 4 and 5 are repeated. Changes are tracked in Change Management Record(s).	Risk Assessment Change Management Record(s) if required	For each functional requirement, the regulated company should assess the risk of software functional failure by identifying severity, probability, and detectability, and calculating an overall risk priority in accordance with Appendix M3 of *ISPE GAMP 5* [2]. To assess probability, the regulated company should consider the complexity of the functionality and the results of the supplier assessment. To assess detectability, the regulated company should consider whether detectability can be increased via in-process or downstream controls. Once the risk assessment is carried out, each functional requirement will have a residual risk quantified by a risk priority of High, Medium, or Low.
6	Identify risk-reduction controls that can be implemented or performed by the IT/IS service provider **Note:** If software changes occur while activity is in progress, then Activities 3 to 6 are repeated. If GxP-intended use is revised while activity is in progress, then Activities 4 to 6 are repeated. If risk assessment is revised while activity is in progress, then Activities 5 and 6 are repeated. Changes are tracked in Change Management Record(s).	Revised Risk Assessment if required Revised Contract or Quality Agreement if required Change Management Record(s) if required	Based on the supplier assessment, the regulated company should identify any additional technical or procedural controls or activities that the IT/IS service provider will agree to implement or perform that will reduce the probability of functional-requirement failure and, hence, reduce their residual risk. For example, the IT/IS service provider might agree to add technical controls, update or enhance processes, or carry out more in-depth testing of a specific high-risk functionality. Once the risk assessment is revisited, each functional requirement will have a residual risk quantified by a risk priority of High, Medium, or Low.

Table 4.4 Recommended Validation Strategy (continued)

Id.	Activity	Evidence	Further Details
7	Perform user acceptance/ configuration testing **Note:** If software changes occur while activity is in progress, then Activities 3 to 7 are repeated. If GxP-intended use is revised while activity is in progress, then Activities 4 to 7 are repeated. If risk assessment is revised while activity is in progress, then Activities 5 to 7 are repeated. Changes are tracked in Change Management Record(s).	User Acceptance/ Configuration Test Evidence Change Management Record(s) if required	The regulated company should decide the test approach for each functional requirement (including configuration) specified in the User Requirements/Configuration Specification. The decision on whether to test, and the rigor of testing, should be based on the risk priority of the functional requirement. The regulated company should use the testing results as input to an ongoing assessment of the effectiveness of the IT/IS service provider's development process. If the testing identifies an unacceptable number of defects, then this points to an ineffective process, which in turn could require the risk assessment to be revisited.
8	Summarize state of compliance with GxP regulations and fitness for intended use and confirm authorization to go live **Note:** If software changes occur while activity is in progress, then Activities 3 to 8 are repeated. If GxP-intended use is revised while activity is in progress, then Activities 4 to 8 are repeated. If risk assessment is revised while activity is in progress, then Activities 5 to 8 are repeated. Changes are tracked in Change Management Record(s).	Validation Report Change Management Record(s) if required	For the initial release of the GxP computerized system, which includes the initial adoption of the SaaS solution, the regulated company should produce a computerized system validation report to summarize the state of compliance with GxP regulations and fitness for intended use and confirm authorization to go live.
9	When software changes occur, carry out activities to maintain compliance with GxP regulations and fitness for intended use **Note:** If GxP-intended use is revised while activity is in progress, then Activities 4 to 7 and 9 are repeated. If risk assessment is revised while activity is in progress, then Activities 5 to 7 and 9 are repeated. Changes are tracked in Change Management Record(s).	User Requirements/ Configuration Specification Risk Assessment User Acceptance/ Configuration Test Evidence Change Management Record(s)	For each software change, Activities 3 to 7 are repeated. The regulated company should also regularly review the production defects that occur as input to an ongoing assessment of the effectiveness of the IT/IS service provider's development and validation process. If the review identifies an unacceptable number of defects, then this points to ineffective processes, which in turn could require the risk assessment to be revisited. In addition, the regulated company should consider carrying out a reassessment of the IT/IS service provider and/or revising their own validation process.

4.12 Managing Data in an Outsourced Environment (IT/IS Service Provider Support to Data Governance)

Data management/governance is:

> *"the planning, execution and oversight of policies, practices and projects that acquire, control, protect, deliver, and enhance the value of data and information assets."* [66]

Effective data governance enables:

- Quality decision-making based on risk
- Protection of business-critical data (at rest and in transmission)
- Management of data integrity
- Improved quality of data
- Robust system and information security
- Strong quality culture

The IT/IS service provider should have appropriate data governance controls to protect customer data. Such controls include:

- Information security to minimize data breaches
- Identity and access management controls to restrict administration and support team access to customer data
- Restricted access based on least privilege
- Backup and restoration controls
- DR controls
- Archive controls
- Data privacy controls

Contract termination and data retrieval should be addressed within the context of data governance and should be documented in a quality agreement with respect to portability, notification, and access for retrieval. Data must be retrievable in a form that is readable or can be migrated to alternative solutions or file formats. This should include relevant metadata. Table 4.5 lists an example of the responsibilities as it relates to the data in an IT outsourced environment.

Table 4.5: Shared Data Integrity Responsibilities

Cloud Service Provider	Regulated User
• Establish security controls to support confidentiality, integrity, and availability of customer data. • Implement robust risk and quality management processes to ensure quality of delivered products and services. • Follow industry best practices for infrastructure control, software development, and service delivery to ensure all components of the cloud remain in a controlled stated. This includes implementing proper governance controls for service management, update management, and review of access controls. • Implement robust data encryption technology to encrypt customer data at rest and in transit.	• Establish governance controls and operational processes covering data integrity, system administration, and proper operational use of the application. • Implement logical security controls and processes to protect against unauthorized access to the cloud application. • Conduct end-user training on proper system use. • Manage data inputs, processing, storage, and outputs for completeness, accuracy, and timeliness, while adhering to the principles of ALCOA+. • Perform system user acceptance testing to verify fitness for intended use and regulatory compliance. • Perform periodic review with cloud service provider input.

4.13 IT Service Operational Considerations

4.13.1 Change Management

For many aspects of cloud service provision, change management is applied by the IT/IS service provider. Without effective change management processes, the implications can be significant. The second most prevalent threat identified within the "2019 Cloud Security Alliance's Cloud Computing Top Threats" report is where computing assets are set up incorrectly (misconfigured), leaving them vulnerable to malicious activity due to ineffective change control processes. [52] This is in part due to the dynamic nature of the cloud computing environment, where resources are provisioned on demand, at speed and at scale.

Service management tools and automation are used in support of the change management process to ensure reliable and consistent control of changes, in particular to support rapid change. Regulated companies need to consider how changes are managed across the entire technology stack that supports the contracted service. For SaaS offerings this needs to encompass all updates or changes to the software, which includes feature/functionality enhancements, defect fixes, and security patches. For IaaS and PaaS services, this should cover all aspects of the infrastructure managed by the IT/IS service provider. Attention should be focused on the automation tools and the techniques employed by the IT/IS service provider to deploy the changes. Monitoring and scanning tools should be in place to detect misconfigured resources, and defined processes should be implemented to remediate issues as soon as they are detected.

Controls integrated into service management tools and deployment tools should ensure appropriate verification and authorization of changes prior to being released to production environments.

4.13.2 Resilience and Availability

Service design resilience enables service availability and should be one of the key risk considerations when considering an IT/IS service provider, especially where the services support critical business processes. Although resilience should be built into the design of the cloud service, outages still occur across multiple cloud service locations. Without robust IT DR and contingency planning, these can result in significant business impact.

IT/IS service provider due diligence should consider how the IT/IS service provider provides a resilient platform through measures such as clustering, replication, and high availability cloud architecture. It should also consider mechanisms that the IT/IS service provider has implemented to protect against potential risks to service disruption (e.g., Distributed Denial of Service (DDOS) attacks), which could render SaaS services unusable to customers.

To mitigate the risk of service failure, IT/IS service providers should have appropriate business continuity and DR plans in place that are periodically tested and meet the regulated company's recovery objectives. Additionally, client data can be safeguarded through real-time replication of data to a secondary location, which can also be supplemented by periodic backup and restore processes. It is important to determine which controls are valid and effective for the services procured. These arrangements should be enforced through MSAs, quality agreements, and SLAs.

Regulated companies should also have business continuity strategies and plans in place to ensure continued operation of critical processes in the event of a prolonged outage.

4.13.3 Patching, Upgrades, Delivery Frequency

Patching (security and functional), upgrades, monitoring, backups, and DR are all part of the standard services offered by a service provider, regardless of regulatory considerations. The regulated company needs to understand the business processes of the IT/IS service provider. The assessment of service provider documentation and applying a risk-based approach to upgrades and patching also assists the regulated company in determining if any additional validation or qualification activities are required of them during upgrades. Be sure to determine and evaluate how the service provider's change cycle provides an acceptable risk profile, in alignment with the intended use of the service. Provisions for receiving data in a portable format, or in intervals, and the subsequent deletion of the regulated company's data should be addressed in a quality agreement, SLA, or another contract between both parties.

Automated IT services and monitoring tools provide services that can be directly leveraged by the regulated company in a proactive approach. Security vulnerabilities, database performance, component failures, application and platform errors can be routinely monitored to provide real-time feedback on the status of the IT services and supporting infrastructure. These items should be outlined in the MSA or quality agreement that delineates the specifics between the companies and the management of the quality product life cycle considerations.

4.14 IT Continuity

Continued or high availability of IT service is critical to the survival of the business as a whole. This can be achieved by introducing a planned and organized set of risk-reduction measures and recovery options: IT Service Continuity Management (ITSCM).

There may be times when an unplanned interruption or a reduction of quality to an IT service occurs. Such events may typically be referred to as incidents and a key element of service continuity management is to implement an incident management process. According to ITIL [33]:

> *"The incident management process ensures that normal service operation is restored as quickly as possible, and the business impact is minimized."*

ITSCM should be viewed as a constituent of the overall business continuity management. The goal of the process is to make sure services are restored and available within agreed-upon business timelines after major service disruptions.

ITIL [33] differentiates between incident management, which handles a range of incidents of varying impact levels, and ITSCM, which is focused on the broader handling and resolving of large-scale disasters. To achieve this, a fundamental area of focus for ITSCM is planning for:

- Incident Prevention

- Incident Prediction
- Incident Management

with the goal of maintaining service availability (in accordance with minimum uptime requirements) and performance at the highest possible levels before, during, and after a disaster-level incident. The objective is to minimize downtime and the financial and business impacts incurred from incidents, by implementing standard procedures and processes for when such disaster-level incidents arise.

Having a well-documented, clear IT Service Continuity Plan helps minimize any delays caused by several factors including steep learning curves, unfamiliarity with the system and business needs, panic, or out-of-hours alerts. It also provides a means for defining what constitutes an incident and what constitutes a disaster. This is important because different service providers (or indeed businesses) may have different definitions. One basic definition of a disaster [33] is:

> *"A sudden unplanned event that causes great damage or serious loss to an organization. A disaster results in an organization failing to provide critical business functions for some predetermine minimum period of time."*

Organizations need to define and document what they see as damage and loss to critical business processes, and for what length of time it occurs.

4.14.1 Relationship to Business Continuity Management

Business Continuity Management (BCM) is a process for managing risks that could seriously affect the business and safeguarding the interests of key stakeholders, as well as critical business activities including patient safety, product quality, and data integrity. It involves reducing risks to an acceptable level and planning for the recovery of business processes should a disruption or disaster affect the business. Some of these risks may be outside of the control of IT, such as a pandemic, natural disasters (e.g., fire/flood/earthquake), or societal, and others, including disaster-level events, may be IT related.

Accordingly, in the wider context, BCM sets the objectives, scope, and requirements for ITSCM. IT teams should work together with the BCM team to identify the potential business impacts of an IT disaster and ensure there are arrangements to prevent and recover from disaster-level events included in the business continuity plan.

To meet the BCM goals stated above, the ITSCM process should:

- Create/manage IT service continuity and recovery plans in the event of an IT disaster
- Liaise with service providers and leverage as appropriate knowledge and experience to minimize the risks and impacts of incidents with their products and services
- Perform ongoing risk and impact analysis and appropriate maintenance/revision of plans

4.14.2 An Outline of an ITSCM Process

There are four recommended process phases to help define and manage IT service continuity plans:

1. Planning
2. Defining clear responsibilities
3. Testing and rehearsing
4. Monitoring and improving

4.14.3 Planning

Commence by considering a number of high-level issues and, depending on the outcome of this risk-based assessment, start to create a plan.

The following is an example list, but which is by no means complete:

- Are there any inherent risks and threats to the business?
- Are there any specific types of disasters that should be planned for?
- Are there any underlying values that need to be adhered to?
- How does the business respond to an incident?
- How will the business respond to each disaster?
- What are the business support systems?
- How does the business maintain/share the information needed to support and restore critical systems?

When preparing plans, consider extreme situations such as a total loss of IT services, for example, due to a widespread cyberattack or the impact of a global pandemic.

4.14.4 Defining Clear Responsibilities

ITSCM should always have a clear sense of roles and responsibilities not only for disasters themselves, but for ongoing monitoring and improvement.

To effectively plan and implement ITSCM practices across the organization, many businesses appoint an IT Service Continuity Manager and an IT Service Continuity Recovery Team.

- **IT Service Continuity Manager –** The IT Service Continuity Manager is responsible for managing risks that could seriously impact IT services. They typically lead the development of continuity plans, manage monitoring and assessment activities, and oversee the plans during a disaster situation.

  They ensure that the IT/IS service provider can provide minimum-agreed service levels in cases of disaster by reducing the risk to an acceptable level and planning for the recovery of IT services.

- **IT Service Continuity Recovery Team –** The IT Service Continuity Recovery Team is led by the IT Service Continuity Manager and participates in the testing, rehearsing, and invocation of the service continuity recovery plan. The team should include technical staff for technical procedures, QA and testing professionals for testing, users for both testing and during invocation, and key representatives from cross-organization departments for communication and coordination of activities.

4.14.5 Testing and Rehearsing

When establishing the plan, it is vital to gain confidence that the plan will deliver the resilience required. This needs to be established before the plan needs to be invoked in a real disaster situation. Discovering during a disruptive event that some elements of the plan do not provide a successful outcome is too late. For this reason, it is important that tests and rehearsals (incident management drills) are performed.

The regular performance of such tests or rehearsals provides inputs and ideas for what could be done better (on existing services) and initiate improvement initiatives. Additionally, this activity informs and educates members of the IT Service Continuity Recovery Team.

4.14.6 Monitoring and Improving

Implementing ITSCM should be viewed as a continuous process, and this is reinforced by the need for ongoing monitoring of the service delivery. Such monitoring activities include, but are not limited to:

- Review of the testing and rehearsal activities
- Review of system backup outcomes
- Review of the system's availability and downtime
- Review of changes and incidents
- Review of preventive maintenance outcomes
- Review of service provider performance

Such activities allow the IT Service Continuity Recovery Team to communicate across the organization and to continue to update the plan, perform further monitoring, and keep the improvement process moving forward.

4.14.7 Typical Plan Content

The plan should be clear, concise, and expect a level of knowledge but not presume explicit expertise, in the event that external assistance is required to rebuild systems. Each procedure should be self-contained so that it can be utilized to effect recovery of a single system or component. Each plan should also contain details of prerequisites. This means that in the event of multiple component failures the correct sequence can be followed.

In summary, the IT Service Continuity Plan should minimally contain:

- Details of the combined component RTOs (Recovery Time Objectives) and RPOs (Recovery Point Objectives)
- An overview of IT architecture
- Roles and responsibilities
- Invocation procedures
- Damage assessment
- Escalation procedures and process flowcharts
- Detailed procedures specifying how to recover each component of the IT system
- Test plans specifying how to test that each component has been recovered successfully
- Incident logs
- Communication and reporting requirements
- Contact details (covering all time zones as necessary)
- Failback procedures
- IT test/rehearsal plan

Additionally, the plan should include detail covering the expected phases to be taken toward resumption of normal service, for example:

- **Initial response:** Damage assessment and invocation of the appropriate incident management teams
- **Service recovery:** This may be staged and offer a degraded service
- **Service delivery in abnormal circumstances:** Interim measures may include relocation of services to another site/region or utilization of redundant or spare equipment. This is a temporary measure to provide a limited service until normal service can be resumed.
- **Normal service resumption:** Returning to the usual service, failback from the abnormal service delivery

Note: Response planning should include gathering evidence of the gap between the point of failure and the last successful backup (i.e., what information may have been lost). This allows identification of potential data loss and communication regarding which, if any, data (records) need to be entered again.

5 Appendix 1 – References

1. ISPE GAMP® Community of Practice, International Society for Pharmaceutical Engineering (ISPE), www.ispe.org www.ispe.org.

2. *ISPE GAMP® 5: A Risk-Based Approach to Compliant GxP Computerized Systems,* International Society for Pharmaceutical Engineering (ISPE), Fifth Edition, February 2008, www.ispe.org.

3. International Society for Pharmaceutical Engineering (ISPE), www.ispe.org.

4. International Council for Harmonisation (ICH), ICH Harmonised Tripartite Guideline, *Quality Risk Management – Q9,* Step 4, 9 November 2005, www.ich.org.

5. US FDA Center for Devices and Radiological Health (CDRH), Case for Quality, Food and Drug Administration (FDA), www.fda.gov.

6. *ISPE GAMP® RDI Good Practice Guide: Data Integrity by Design,* International Society for Pharmaceutical Engineering (ISPE), First Edition, October 2020, www.ispe.org.

7. *ISPE GAMP® Guide: Records and Data Integrity,* International Society for Pharmaceutical Engineering (ISPE), First Edition, March 2017, www.ispe.org.

8. Critical Thinking as Defined by the National Council for Excellence in Critical Thinking, 1987 – A statement by Michael Scriven & Richard Paul, presented at the 8th Annual International Conference on Critical Thinking and Education Reform, Summer 1987, https://www.criticalthinking.org/pages/defining-critical-thinking/766.

9. PIC/S Guidance: PI 041-1 Good Practices for Data Management and Integrity in Regulated GMP/GDP Environments, 1 July 2021, Pharmaceutical Inspection Co-operation Scheme (PIC/S), www.picscheme.org.

10. PIC/S Guidance: PI 011-3 Good Practices for Computerised Systems in Regulated "GXP" Environments, 25 September 2007, Pharmaceutical Inspection Co-operation Scheme (PIC/S), www.picscheme.org.

11. *ISPE GAMP® RDI Good Practice Guide: Data Integrity – Key Concepts,* International Society for Pharmaceutical Engineering (ISPE), First Edition, October 2018, www.ispe.org.

12. *ISPE GAMP® RDI Good Practice Guide: Data Integrity – Manufacturing Records,* International Society for Pharmaceutical Engineering (ISPE), First Edition, May 2019, www.ispe.org.

13. Wingate, G., ed., *Pharmaceutical Computer Systems Validation: Quality Assurance, Risk Management and Regulatory Compliance,* 2nd Edition, CRC Press, February 2010.

14. *ISPE GAMP® Good Practice Guide: A Risk-Based Approach to Testing of GxP Systems,* International Society for Pharmaceutical Engineering (ISPE), Second Edition, December 2012, www.ispe.org.

15. ISTQB, "Certified Tester Foundation Syllabus Version 2018 V3.1," ISTQB® (International Software Testing Qualifications Board), November 2019, [Online], Accessed December 2020, www.istqb.org.

16. Bezier, B., Software Testing Techniques, 2nd Edition, Van Nostrand Reinhold, June 1990.

17. *ISPE GAMP® Good Practice Guide Series,* International Society for Pharmaceutical Engineering (ISPE), www.ispe.org.

18. *ISPE GAMP® Good Practice Guide: A Risk-Based Approach to Operation of GxP Computerized Systems,* International Society for Pharmaceutical Engineering (ISPE), First Edition, January 2010, www.ispe.org.

19. EudraLex Volume 4 – Guidelines for Good Manufacturing Practices for Medicinal Products for Human and Veterinary Use, Annex 11: Computerized Systems, June 2011, http://ec.europa.eu/health/documents/eudralex/vol-4/index_en.htm.

20. EMA, "Q&A: Good clinical practice (GCP)," European Medicines Agency (EMA), Accessed 16 June 2021, https://www.ema.europa.eu/en/human-regulatory/research-development/compliance/good-clinical-practice/qa-good-clinical-practice-gcp.

21. Shitamoto, K. and Gurumoorthi, S., "Understanding FDA's CSA Guidance in the Context of Current Regulations and GAMP®," *American Pharmaceutical Review,* 29 March 2021, https://www.americanpharmaceuticalreview.com/Featured-Articles/574659-Understanding-FDA-s-CSA-Guidance-in-the-Context-of-Current-Regulations-and-GAMP.

22. Black, R., *Advanced Software Testing,* Vol. 1, Rocky Nook, October 2010.

23. "Manifesto for Agile Software Development," 2001, https://agilemanifesto.org.

24. The Scrum Framework Poster, Scrum.org, accessed 20 August 2021, www.scrum.org.

25. Wyn, S., Reid, C.J., Clark, C., Rutherford, M.L., Watson, H.D., Vuolo-Schuessler, L.L., Perez, A., "Why ISPE GAMP® Supports the FDA CDRH: Case for Quality Program," *Pharmaceutical Engineering,* November/December 2019, Vol. 39, No. 6, pp. 37-41, www.ispe.org.

26. "Definition of Agile ceremonies," Webopedia, Updated: 24 May 2021, https://www.webopedia.com/definitions/agile-ceremonies.

27. McDonagh, A., Dubovik, S., Cherry, M.R., O'Brien, D., Jones, S., "Agile Software Development in GxP Regulated Environments GAMP® Special Interest Group," iSpeak Blog, 3 August 2020, International Society for Pharmaceutical Engineering (ISPE), www.ispe.org.

28. Emergen, www.emergn.com.

29. Wake, B., "INVEST in Good Stories, and SMART Tasks," XP 123 Exploring Extreme Programming, 17 August 2003, https://xp123.com/articles/invest-in-good-stories-and-smart-tasks.

30. 21 CFR Part 11 – Electronic Records; Electronic Signatures, Code of Federal Regulations, US Food and Drug Administration (FDA), www.fda.gov.

31. Speer, J., "FDA Case for Quality: 2018 Comprehensive Review," *FDA Regulations and True Quality and Regulatory Affairs and Quality Management System (QMS) and Manufacturing and Computer System Validation,* 1 January 2019, https://www.greenlight.guru/blog/fda-case-for-quality-2018-comprehensive-review.

32. Hern, A., "The two-pizza rule and the secret of Amazon's success," The Guardian, 24 April 2018, https://www.theguardian.com/technology/2018/apr/24/the-two-pizza-rule-and-the-secret-of-amazons-success.

33. ITIL® Foundation, ITIL 4 Edition, London, UK:Axelos, 2019, www.axelos.com.

34. COBIT (Control Objectives for Information and Related Technology), ISACA, www.isaca.org.

35. ISO 9001:2015 Quality Management Systems – Requirements, International Organization for Standardization (ISO), www.iso.org.

36. ISO/IEC 27001:2013 Information Technology -- Security Techniques -- Information Security Management Systems -- Requirements, ISO/IEC JTC1, International Organization for Standardization (ISO), www.iso.org, and International Electrotechnical Commission (IEC), www.iec.ch.

37. National Institute of Standards and Technology (NIST), www.nist.gov.

38. Cloud Security Alliance (CSA), https://cloudsecurityalliance.org.

39. Sarbanes-Oxley Act of 2002, US Securities and Exchange Commission (SEC), http://www.sec.gov/about/laws/soa2002.pdf.

40. Reid, C. and Wyn, S., "IT Services: Applying Good IT Practice & Automation," *Pharmaceutical Engineering,* May/June 2021, Vol. 41, No. 3, pp. 14-17, www.ispe.org.

41. Regulation (EU) 2016/679 of the European Parliament and of the Council of 27 April 2016 on the protection of natural persons with regard to the processing of personal data and on the free movement of such data, and repealing Directive 95/46/EC (General Data Protection Regulation), GDPR Regulation (EU) 2016/679, (General Data Protection Regulation), https://gdpr-info.eu.

42. California Consumer Privacy Act (CCPA) of 2018, https://oag.ca.gov/privacy/ccpa.

43. NIST, *Framework for Improving Critical Infrastructure Cybersecurity,* Version 1.1, National Institute of Standards and Technology (NIST), 16 April 2018, www.nist.gov.

44. EMA, EMA/226170/2021 (draft) Guideline on computerised systems and electronic data in clinical trials, Good European Clinical Practice Inspectors Working Group (GCP IWG), European Medicines Agency (EMA), https://www.ema.europa.eu/en/documents/regulatory-procedural-guideline/draft-guideline-computerised-systems-electronic-data-clinical-trials_en.pdf.

45. "NIST Releases Version 1.1 of its Popular Cybersecurity Framework," National Institute of Standards and Technology (NIST), Released 16 April 2018, Updated 5 March 2021, www.nist.gov.

46. 21 CFR Part 211 – Current Good Manufacturing Practice for Finished Pharmaceuticals, Code of Federal Regulations, US Food and Drug Administration (FDA), www.fda.gov.

47. 21 CFR Part 820 – Quality System Regulation; Code of Federal Regulations, US Food and Drug Administration (FDA), www.fda.gov.

48. ISO 22301:2019 Security and resilience — Business continuity management systems — Requirements, International Organization for Standardization (ISO), www.iso.org.

49. *ISPE GAMP® Series,* International Society for Pharmaceutical Engineering (ISPE), www.ispe.org.

50. The Federal Assembly of the Swiss Confederation, Federal Act on Data Protection (FADP), of 19 June 1992 (Status as of 1 March 2019), https://www.fedlex.admin.ch/eli/cc/1993/1945_1945_1945/en.

51. Guide to the General Data Protection Regulation (GDPR), Information Commissioner's Office (UK), 22 March 2018, https://www.gov.uk/government/publications/guide-to-the-general-data-protection-regulation.

52. Cloud Security Alliance, "Top Threats to Cloud Computing, The Egregious 11," Cloud Security Alliance (CSA), April 2020, https://cloudsecurityalliance.org.

53. National Institute of Standards and Technology (NIST) Computer Security Resource Center, https://csrc.nist.gov.

54. *ISPE GAMP® Good Practice Guide: IT Infrastructure Control and Compliance,* International Society for Pharmaceutical Engineering (ISPE), Second Edition, August 2017, www.ispe.org.

55. ISACA, www.isaca.org.

56. ISO 14971:2019, Medical devices — Application of risk management to medical devices, International Organization for Standardization (ISO), www.iso.org.

57. Committee of Sponsoring Organizations of the Treadway Commission (COSO), www.coso.org.

58. Perez, A.D., Canterbury, J., Hansen, E., Samardelis, J. S., Longden, H., Rambo, R. L., "Application of the SOC 2+ Process to Assessment of GxP Suppliers of IT Services," *Pharmaceutical Engineering,* July/August 2019, Vol. 39, No. 4, pp. 14-20, www.ispe.org.

59. ISO/IEC DIS 27002, Information security, cybersecurity and privacy protection — Information security controls, International Organization for Standardization (ISO), www.iso.org.

60. ISO/IEC 27017:2015, Information technology — Security techniques — Code of practice for information security controls based on ISO/IEC 27002 for cloud services, International Organization for Standardization (ISO), www.iso.org, and International Electrotechnical Commission (IEC), www.iec.ch.

61. ISO/IEC 20000-1:2018 Information technology — Service management — Part 1: Service management system requirements, ISO/IEC JTC1, International Organization for Standardization (ISO), www.iso.org, and International Electrotechnical Commission (IEC), www.iec.ch.

62. Simmon, E., NIST, Special Publication 500-322, "Evaluation of Cloud Computing Services Based on NIST SP 800-145," NIST Cloud Computing Cloud Services Working Group, NIST Cloud Computing Program, Information Technology Laboratory, National Institute of Standards and Technology (NIST), February 2018, https://csrc.nist.gov.

63. Security Trust Assurance and Risk (STAR) Registry, Cloud Security Alliance (CSA), https://cloudsecurityalliance.org.

64. American Institute of CPAs (AICPA), www.aicpa.org.

65. AICPA, "2017 Trust Services Criteria for Security, Availability, Processing Integrity, Confidentiality, and Privacy," American Institute of CPAs (AICPA), March 2020, www.aicpa.org.

66. DAMA International, DAMA International's Guide to the Data Management Body of Knowledge (DAMA-DMBOK2), First Edition, ISBN, PDF 9780977140084, DAMA International, 2009, https://technicspub.com/dmbok.

6 Appendix 2 – Glossary

6.1 Acronyms and Abbreviations

AI	Artificial Intelligence
AICPA	American Institute of Certified Public Accountants
ALCOA	Attributable, Legible, Contemporaneous, Original, Accurate
ALCOA+	ALCOA, with the addition of Complete, Consistent, Enduring, Available
APAC	Asia Pacific
API	Application Programming Interface
BCM	Business Continuity Management
BDD	Behavior-Driven Development
BI	Business Intelligence
CAIQ	Consensus Assessment Initiative Questionnaire
CAPA	Corrective and Preventive Action
CCPA	California Consumer Privacy Act (US)
CDM	Clinical Data Management
CDRH	Center for Devices and Radiological Health (US)
CFR	Code of Federal Regulations (US)
CI	Continuous Integration
CMDB	Configuration Management Database
COBIT®	Control Objectives for Information and Related Technologies (ISACA)
CoP	Community of Practice
COSO	Committee of Sponsoring Organizations of the Treadway Commission
COTS	Commercial off-the-Shelf
CPA	Certified Public Accountants
CPU	Central Processing Unit
CRM	Customer Relationship Management
CRO	Clinical Research Organization
CSA	Computer Software Assurance
DBMS	Database Management System
DDOS	Distributed Denial of Service
DoD	Definition of Done
DR	Disaster Recovery
EMA	European Medicines Agency

EMEA	Europe, Middle East, Africa
ERP	Enterprise Resource Planning
eTMF	electronic Trial Master File
EU	European Union
FDA	Food and Drug Administration (US)
GDPR	General Data Protection Regulation (EU)
IaaS	Infrastructure as a Service
IaC	Infrastructure as Code
ICFR	Internal Control over Financial Reporting
ICH	International Council for Harmonisation
ID	Identification
IS	Information System
IT	Information Technology
ITSCM	IT Service Continuity Management
LDAP	Lightweight Directory Access Protocol
MSA	Master Service Agreement
MVP	Minimum Viable Product
NIST	National Institute of Standards and Technology (US)
OS	Operating System
OSS	Open-Source Software
PaaS	Platform as a Service
PAT	Process Analytical Technology
PBI	Product Backlog Items
PPI	Personal and Private Information
PQS	Pharmaceutical Quality System
QA	Quality Assurance
Q&A	Question and Answer
QMS	Quality Management System
QRM	Quality Risk Management
RPO	Recovery Point Objective
RTO	Recovery Time Objective
SaaS	Software as a Service
SaMD	Software as a Medical Device
SLA	Service Level Agreement

SME	Subject Matter Expert
SOP	Standard Operating Procedures
STAR	Security Trust Assurance and Risk
UAT	User Acceptance Testing
URS	User Requirements Specification
US	United States
XaaS	Infrastructure/Platform/Software as a Service

6.2 Definitions

Ad Hoc Testing (*ISPE GAMP® RDI Good Practice Guide: Data Integrity by Design* [6])

Ad hoc testing is unscripted testing performed without planning or pre-defined documentation. It is aimed at finding defects as early as possible

Agile Software Development

Software development models and methods based on iterative, incremental, and exploratory approaches, where requirements and solutions evolve through collaboration between self-organizing cross-functional teams.

Computerized System (*ISPE GAMP® 5* [2])

A broad range of systems including, but not limited to, automated manufacturing equipment, automated laboratory equipment, process control and process analytical, manufacturing execution, laboratory information management, manufacturing resource planning, clinical trials data management, vigilance and document management systems. The computerized system consists of the hardware, software, and network components, together with the controlled functions and associated documentation.

Critical Thinking (*ISPE GAMP® Guide: Records and Data Integrity* [7])

A systematic, rational, and disciplined process of evaluating information from a variety of perspectives to yield a balanced and well-reasoned answer.

Error Guessing (*ISPE GAMP® RDI Good Practice Guide: Data Integrity by Design* [6])

Error guessing is a testing technique designed to expose anticipated and potential defects based on the specialist tester's knowledge and experience of failure modes.

Exploratory Testing (*ISPE GAMP® RDI Good Practice Guide: Data Integrity by Design* [6])

Exploratory testing is experience-based testing where the tester spontaneously designs and executes tests based on existing specialist tester's knowledge and experience, prior exploration of test item (including results from previous tests), and typical common software behaviors and types of failure and defects.

Quality Risk Management (QRM) (ICH Q9 [4])

A systematic process for the assessment, control, communication and review of risks to the quality of the drug (medicinal) product across the product lifecycle.

Unscripted Testing (*ISPE GAMP® RDI Good Practice Guide: Data Integrity by Design* [6])

Unscripted testing is testing in which, while the testing activity is still documented, the tester's actions are not prescribed by detailed instructions in advance of test execution to the same extent as scripted testing. Details of the testing performed, by whom, and outcomes and conclusions are still recorded in all cases.

Waterfall Model

A model used in the system development life cycle to develop a system using a linear and sequential approach. The model is divided into different phases and the output of one phase is used as the input to the next phase.

600 N. Westshore Blvd., Suite 900, Tampa, Florida 33609 USA
Tel: +1-813-960-2105, Fax: +1-813-264-2816

www.ISPE.org