Reprinted from
PHARMACEUTICAL ENGINEERING®
The Official Journal of ISPE
May/June 2003, Vol. 23 No. 3

Risk Assessment for Use of Automated Systems Supporting Manufacturing Processes

Part 1 - Functional Risk

by the ISPE GAMP Forum

This article illustrates the risk analysis guidance discussed in GAMP 4.[5] By applying GAMP's risk analysis method to three generic classes of software systems, this article acts as both an introduction to the method and an illustration of its use.

The FDA recently announced a significant new initiative to enhance US regulation of pharmaceutical manufacturing and product quality.[1,2] The initiative is based on the FDA's current Good Manufacturing Practice (cGMP) program and covers veterinary and human drugs, including human biological drug products, such as vaccines. The aim is to enhance the established 'quality systems' approach with risk management. Other regulatory authorities have already embraced science-based risk management as a key operating principle.[3,4] With this in mind, this article endeavors to develop a common understanding of the relative risks posed by different types of automated system used to support manufacturing processes. An underlying assumption is that the rigor of validation for an automated system should be commensurate to risk. The significance of any compliance deficiency then

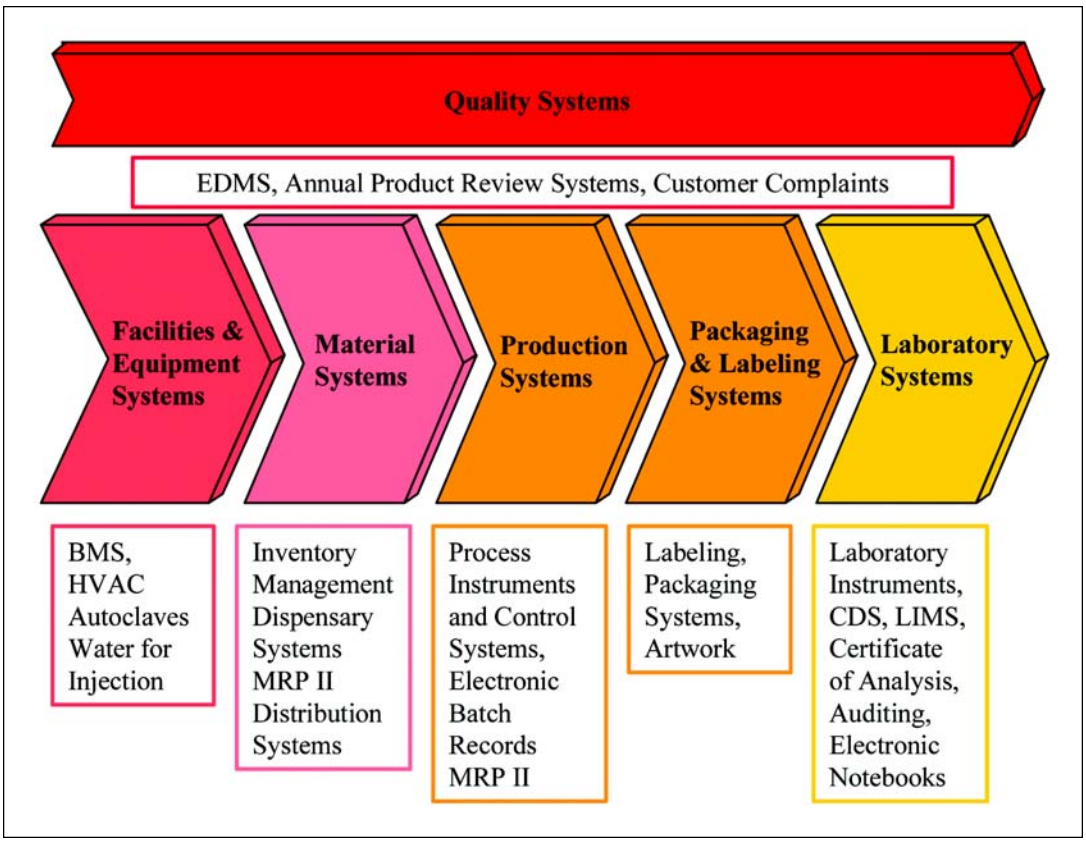

Figure 1. Use of automated systems.

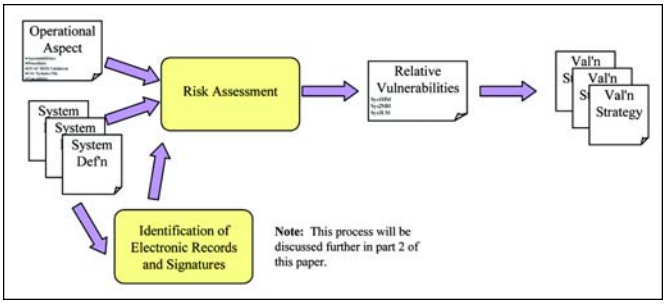

Figure 2. Risk assessment process.

needs to take account of the use of that system in supporting a manufacturing process.

This analysis of relative risks is split into two parts:

- The first part concentrates on functional risks associated with different classes of software solution.
- The second part, to be published later this year, will address the relative risks associated with electronic records.

This article illustrates the risk analysis guidance discussed in GAMP 4.[5] By applying GAMP's risk analysis method to three generic classes of software systems, this article acts as both an introduction to the method and an illustration of its use.

Use of Automated Systems

Automated systems are widely used in support of pharmaceutical manufacturing. A workflow analysis of the manufacturing process (based on the FDA's Systems Approach to inspection[6]) identifies six main operational aspects where computer systems are used:

- Quality Systems - dealing with roles and procedural controls
- Facilities and Equipment Systems - dealing with the physical environment used in the production of drug products
- Materials Systems - dealing with drug product components, inventory control processes, and drug storage
- Production Systems - dealing with manufacturing controls
- Packaging and Labeling Systems - dealing with packaging and labeling
- Laboratory Systems - dealing with analytical testing

Figure 1 illustrates where various automated systems might be used. It is important to appreciate that some automated systems support multiple aspects of the manufacturing process such as MRP II systems, while other automated systems are dedicated to specific aspects of the process such as HPLC systems.

Risk Assessment Process

1. The first step of the risk assessment process used here uses the six operational aspects of the manufacturing process to identify the functional criticality of an automated system.
2. The second step is an analysis of the automated system's vulnerability to deficient operation.
3. The third step is the determination of a validation strategy. Differing levels of system vulnerability require different levels of rigor of validation activity.

Equally, validation must address any electronic record/signature requirements. The three-step risk assessment process is illustrated in Figure 2.

Functional Criticality

Determining which operational aspects of the manufacturing process that are most critical requires an understanding of the potential impact that these aspects have on drug product safety, quality, and efficacy. The Canadian Health Products and Food Branch Inspectorate have already identified a number of high risk issues that are likely to result in non-compliant drug product and present an immediate or latent public health risk.[4] These high-risk issues are applied here to automated systems and aligned to the six operational areas identified previously.

Quality Systems

- Document Management
- SOP Administration
- Security Access Controls (e.g., User Profiles and Password Management)
- Change Control Records
- Customer Complaints
- Adverse Event Reporting
- Review/Audit/Corrective Actions Management
- Training Records

Facilities and Equipment Systems

- HVAC Controls and Alarm Handling
- Critical Equipment and Instrumentation (Calibration and Maintenance)
- Change Control Records
- Validation Records

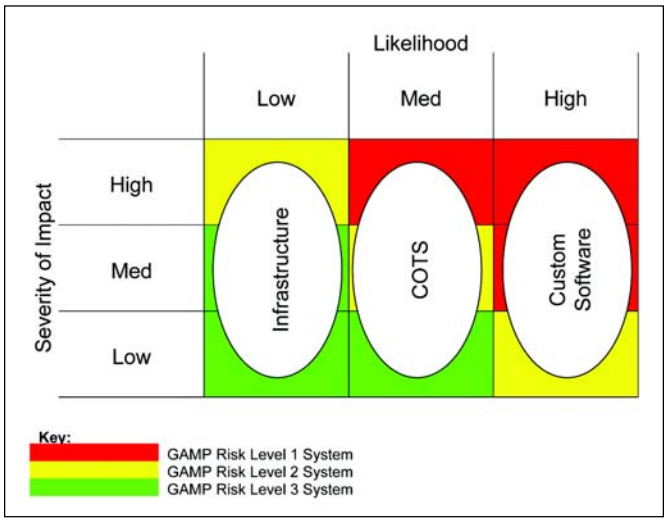

Figure 3. GAMP risk classifications.

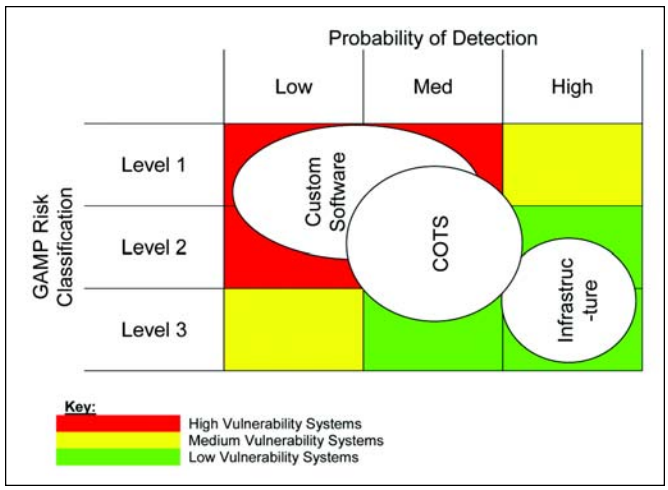

Figure 4. Relative system vulnerabilities.

Materials Systems

- Traceability of Material Handling
- Raw Material Inspection/Testing/Status Management
- Storage conditions
- Containers Usage and Cleaning Management
- Distribution Records and Recall Management

Production Systems

- Recipe/Formulation Management
- Batch Manufacturing Instruction and Records
- In-Process Testing
- Yield Calculation
- Purified Water
- Aseptic Filling

Packaging and Labeling Systems

- Labeling Information

Laboratory Systems

- QC Raw Data
- Stability Testing
- Sterility Testing
- QC Analytical Results
- Quality Disposition
- Out of Specification Investigations

The rigor of validation for automated systems supporting these critical operational aspects of the manufacturing process should take account of their composite custom (bespoke) software, commercial Off-The-Shelf (COTS) software, and supporting computer network infrastructure.

System Vulnerability

GAMP's Risk Assessment methodology[5] is used here to analyze the relative vulnerabilities of three typical classes of software system:

- *Custom Software* refers to a software solution that has been specifically developed for application within a pharmaceutical manufacturing set of requirements (see GAMP 4 glossary of terms). It reflects GAMP Software Category 5 - 'Custom (bespoke) Software' or the application specific configuration code of a GAMP Software Category 4 - 'Configurable Software Packages' system.

- *Commercial Off-The-Shelf Software (COTS)* refers to existing (i.e., not developed specifically for an application) standard software products used across many applications within the pharmaceutical operations and potentially other industries. It reflects GAMP Software Category 3 - 'Standard Software Packages' or GAMP Software Category 1 - 'Operating Systems' or the standard product component of a GAMP Software Category 4 - 'Configurable Software Packages' system.

- *Infrastructure* refers to the typical infrastructure consisting of physical network components, switches, hubs, routers, servers, firewalls, network operating systems, and their configuration.

Initially, the three classes of automated system are analyzed, based on how significant a threat arising from the system might be, both in terms of system function, and system data - *Figure 3*. With all three classes of system, the severity of impact that may arise from the system will depend on its application (i.e., number of critical operational aspects of the manufacturing process the system supports, what breadth of business operations it impacts, and to what extent the system might fail). Each class of system may therefore represent a threat with low, medium, or high severity. However, the likelihood of failure will vary with class of system.

Custom Software

These systems have been developed specifically for this application. This application will, therefore, be the first use of the software so it will not have been proven through an installed base. This class of system will, therefore, tend to present a relatively high *Likelihood* of failure. Applying a high *Likelihood* to the GAMP Risk Classification grid therefore classifies *Custom Software* as predominantly a Level 1 or Level 2 risk.

COTS

These systems typically have an existing significant installed base. The software will, therefore, be in part proven by previous validation exercises and by use. However, the likelihood of failure is not insignificant, as these are often highly complex systems that are highly configurable so that parts of the code might be unproven. This class of system will, therefore, tend to present a medium *Likelihood* of failure. Applying a medium *Likelihood* classifies *COTS* as a Level 1, Level 2, or Level 3 risk.

Infrastructure

The infrastructure is typically built from industry standard network components. These components are proven across all industries as highly robust and also self-correcting (e.g.,

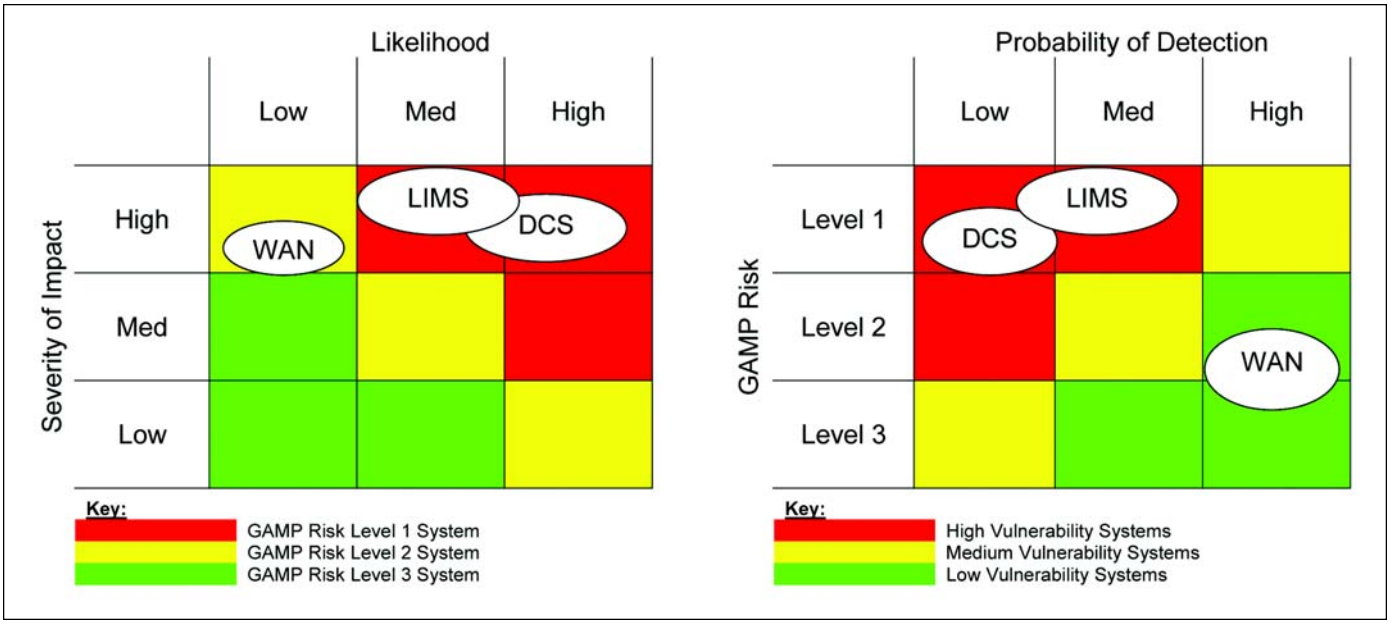

Figure 5. Illustrative systems' vulnerabilities.

TCP/IP protocol). Component failure can often be tolerated without significant impact on infrastructure function or performance. This class of system will, therefore, tend to present a relatively low *Likelihood* of failure. Applying a low *Likelihood* classifies *Infrastructure* as a Level 2, or predominantly, a Level 3 risk.

The relative vulnerability of a system is then deduced by comparing the system's risk classification (Level 1, 2, or 3) with the probability of detecting failure arising from the system - Figure 4. The *Probability of Detection* of failures arising from a system depends on a number of factors, such as:

- error detection function built-into the software function itself
- use of separate and independent systems to duplicate certain functions (redundancy) or monitor the output of the system and report deviations
- use of manual inspections or testing to monitor the correct behavior of the system

Clearly, these last two items will depend on the application, rather than the class of software. However, these different classes of software do tend to have different levels of error detection capabilities:

Custom Software

Error detection is often fairly complex and expensive to develop. It is, therefore, relatively unlikely that a *Custom Software* solution will have good error detection support. These systems will, therefore, tend to have low or medium *Probability of Detection*, yielding a system with a predominantly high vulnerability.

COTS

As COTS have a larger installed base; and therefore, a larger development budget than a *Custom Software* solution, the probability of a COTS product featuring some form of error detection mechanism is higher than with *Custom Software.* These systems will tend to have a mainly medium *Probability of Detection,* yielding a high, medium, or low system vulnerability.

Infrastructure

Most standard network components now have some form of error detection mechanism (e.g., - collision detection at the ethernet level, datagram checksums on TCP/IP). While the correct function of an infrastructure will be largely undetectable to human eyes, these built-in detection mechanisms make it extremely unlikely that an error will be propagated by the infrastructure without detection by the infrastructure itself. In the event of significant infrastructure failure, the applications that employ the infrastructure typically will either report the fault or completely fail, i.e., crash so the failure cannot go undetected. This yields a low system vulnerability.

Rigor of Validation

Broadly, the three classes of software system from *Infrastructures* to *Custom Software* represent increasing vulnerability for public health from drug safety, quality, and efficacy. With increasing vulnerability goes the demand for greater rigor in system validation. Table A lists these classes of risk with suggestions of appropriate levels of compliance activity required to validate that system.

Illustrative Examples

As an illustration, the severity of risk (GAMP Risk Analysis Method step 1) is considered for three typical systems that between them include aspects of each of the classes of software system discussed above.

Distributed Control System (DCS)

While almost certainly based around a proven software DCS product or suite of products, the engineering of DCS installation that controls batch manufacture of a pharmaceutical API is dominated by the application specific configuration and coding. This 'control application' within the DCS will, therefore, fit into the category of Custom Application.

Laboratory Information System (LIMS)

There are now well-established LIMS products on the market that provide the full breadth of function required for information management in most GMP laboratories. As a large part of a typical installation's required functionality is met by standard function, a LIMS can usually be considered as a GAMP Category 4 solution, i.e., a composite of COTS and application specific configuration.

Company Wide Area Network (WAN)

Almost all multisite organizations have some form of WAN. WANs are clearly infrastructure systems, and may include standard hardware and software components such as domain servers, bridges, routers, and firewalls.

Class of System	Vulnerability/ Validation Rigor	Emphasis of User Validation Activities		
		Plan/Report	Design Phases	Qualification Phases
Custom Software Application	High	• Validation Plan and Report • Development SOPs • Supplier Audit with closure on significant deficiencies • Project Audit(s) • Periodic Review • Change Control	• URS (business and regulatory needs) • FS (full functionality of the system) • Design down to the level of module specifications • Design Review process • Source Code Review (general coding practices and detailed walk-through of highest risk code) • Traceability Matrix (comprehensive)	• Detailed risk assessment against the operational aspects of the manufacturing process identified in this article • Comprehensive positive functional testing (it does what it should do) • Risk-focused negative functional testing (it does not do what it should not do where the risk assessment identified vulnerability)
COTS Application	Medium	• Validation Plan and Report • Development SOPs • Supplier Audit with compensating actions for significant deficiencies • Periodic Review • Change Control	• URS (business and regulatory needs) • FS (full functionality for application specific requirements, points to standard product documentation for standard functions) • Design documents application configuration aspects only • Design Review process • Traceability Matrix (user documents to standard product documents).	• High level risk assessment against the operational aspects of the manufacturing process identified in this article • Positive functional testing of the defined user operation for this specific application (it does what it should do) • Risk-focused negative functional testing (it does not do what it should not do where the risk assessment identified vulnerability)
Infrastructure	Low	• SLA • Quality and Compliance Plan • Work SOPs • Periodic Review • Change Control	• Network topology diagram • Network definition (list of supported applications, network performance and security requirements only) • Design (network configuration)	• High level risk assessment against the operational aspects of the manufacturing process identified in this article • Risk-focused functional testing (e.g., security controls, data integrity, backup and recovery)

Increasing vulnerability, increasing need for rigor in validation

Table A. Summary of vulnerabilities and required validation rigor.

Risk Area	High Risk Issues		
	Illustrative DCS	Illustrative LIMS	Illustrative WAN
Quality Systems	-	-	• Security Access Control
Facility and Equipment Systems	-	-	-
Materials Systems	-	• Raw Materials Testing and Status Management	-
Production Systems	• Recipe Formulation and Management • Batch Manufacturing	• In-process testing	-
Packaging and Labeling Systems	-	-	-
Laboratory Systems	-	• QC raw data • QC Analytical results	-

Table B. Illustrative high risk functions for the illustrative systems.

Step 1 - Severity of Risk

The precise role and related risks of DCS, LIMS, and WAN installations will vary from installation to installation. For the purpose of this illustration, Table B suggests some typical functions that each system may provide and can be identified as high-risk issues.

Table B shows that all three of our example systems include high-risk function, and should therefore, be considered high-risk systems. However, this table also helps clarify the severity of the risks relative to each other. LIMS, impacting five different high-risk issues across three of the FDA's inspection systems clearly represents the most severe potential risk to public health.

Steps 2 and 3 - Overall Vulnerability

Assuming that the arguments around the Likelihood and Probability of detection discussed for Custom Software, COTS, and Infrastructure discussed above stand for these three illustrative systems, then application of GAMP's Risk Analysis method steps 2 and 3 will yield relative vulnerabilities as depicted in Figure 5.

The combined steps 1, 2, and 3 of GAMP's functional risk analysis method indicates that both the DCS and the LIMS are high vulnerability systems, and therefore, should be subjected to the full validation rigor proposed in Table A. On the other hand, WAN is a relatively low vulnerability system, and need therefore, only be subjected to validation rigor commensurate with its vulnerability.

Conclusion

This article has applied a functional risk assessment method to the use of automated systems supporting manufacturing processes. It has been shown that functional risk assessment provides a mechanism for assessing and ranking the risks arising from computerized systems. By linking degree of rigor of validation to the overall vulnerability of a system, a process for developing risk-appropriate validation strategies has been demonstrated. High-risk operational aspects of the manufacturing process relative to the use of automated systems have been identified based on previous work by regulatory authorities. The relative risk posed by custom applications, COTS applications and infrastructure also has been analyzed to show the lower vulnerability of infrastructure to erroneous operation impacting drug product quality, efficacy, and safety.

Care must be taken when applying the general risk assessment presented in this article to individual automated systems. It is acknowledged that each system is different. Nevertheless, the general approach is well founded and should help pharmaceutical manufacturers and regulatory authorities alike appreciate the relative rigor of validation appropriate to specific automated systems.

A second part to the article considering the relative risks of electronic records will be published later this year.

References

1. U.S. FDA (2002), Pharmaceutical cGMPs for the 21st Century: A Risk Based Approach, FDA News, 21 August, www.fda.gov.
2. European Union Guide to Directive 91/356/EEC (1991), European Commission Directive Laying Down the Principles of Good Manufacturing Practice for Medicinal Products for Human Use.
3. Trill, A. J., Computerised Systems and GMP - Current Issues, Presentation UK Medicines Control Agency Seminar 'Top 10 GMP Inspection Issues' 24 September 2002.
4. Canadian Health Products and Food Branch Inspectorate (2000), Good Manufacturing Practices - Risk Classification for GMP Observations.
5. ISPE (2001), GAMP Guide for Validation of Automated Systems (known as GAMP 4), International Society for Pharmaceutical Engineering (www.ispe.org).
6. FDA (2002), CPG 7356.002 Drug Manufacturing Inspections: Systems Based Approach.

Acknowledgements

The ISPE GAMP Forum would like to acknowledge the contributions of the GAMP Europe and GAMP Americas Steering Committees in the preparation of this article. In particular, Guy Wingate and Sam Brooks are thanked for developing the founding draft of this work.