



Final Document

IMDRF/SaMD WG/N81 FINAL: 2025

Characterization Considerations for Medical Device Software and Software- Specific Risk

AUTHORING GROUP

IMDRF Software as a Medical Device Working Group

Preface

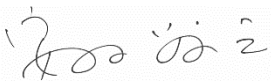
© Copyright 2024 by the International Medical Device Regulators Forum.

This work is copyright. Subject to these Terms and Conditions, you may download, display, print, translate, modify and reproduce the whole or part of this work for your own personal use, for research, for educational purposes or, if you are part of an organisation, for internal use within your organisation, but only if you or your organisation do not use the reproduction for any commercial purpose and retain all disclaimer notices as part of that reproduction. If you use any part of this work, you must include the following acknowledgement (delete inapplicable):

“[Translated or adapted] from [insert name of publication], [year of publication], International Medical Device Regulators Forum, used with the permission of the International Medical Device Regulators Forum. The International Medical Device Regulators Forum is not responsible for the content or accuracy of this [adaption/translation].”

All other rights are reserved and you are not allowed to reproduce the whole or any part of this work in any way (electronic or otherwise) without first being given specific written permission from IMDRF to do so. Requests and inquiries concerning reproduction and rights are to be sent to the IMDRF Secretariat.

Incorporation of this document, in part or in whole, into another document, or its translation into languages other than English, does not convey or represent an endorsement of any kind by the IMDRF.



Naoyuki YASUDA, IMDRF Chair

Contents

1. Introduction	4
2. Purpose and Scope	7
2.1. Purpose of the document	7
2.2. Scope of the document	7
<hr/>	
3. References	9
4. Medical Device Software Characterization Considerations	10
4.1. Intended Use/Intended Purpose Statement	11
4.2. Medical Device Software Description	11
<hr/>	
5. Medical Device Software Risk Characterization	20
5.1. Identification and Analysis	22
5.2. Estimation	22
5.3. Approaches for Risk Categorization	23
<hr/>	
6. Considerations for Implementation	25
Appendix A: Sample Intended Use/Intended Purpose Statement	26
Appendix B: Characterization Feature Summary Table	27
Appendix C: Example Considerations to Understand Software Hazards Associated with Device Design and Intended Use	31
Appendix D: Examples of Linking Characterization Features and Risk	36
Appendix E: Examples Comparing Specific Risk Considerations	44

1. Introduction

Software's role in healthcare is becoming increasingly critical, as a diverse array of products serves various medical and administrative functions across clinical and private settings. A subset of software that is used in healthcare is regulated as a medical device globally by regulatory authorities.

In 2013, the International Medical Devices Regulators Forum (IMDRF) introduced the concept of Software as a Medical Device (SaMD) and subsequently proposed a possible risk categorization framework (IMDRF/SaMD WG/N12 FINAL:2014). Building on the collective experience of its members, the IMDRF SaMD Working Group (WG) now has an opportunity to add to those initial concepts by providing guidance related to device characterization and risk characterization, for a broadened scope of medical device software. In the context of this document, risk characterization is meant to help identify potential harms associated with a medical device software and is based on a careful review of device characterization. Risk characterization can help to develop a robust understanding of the overall risk of the device and can be helpful as an input to risk assessment and management activities or as an input to risk categorization and device classification. This new document is intended to focus on characterization and can supplement categorization/classification frameworks (e.g., N12 and other legally defined classification schemes across jurisdictions) by providing additional considerations on medical device software and related risk characterization. Although this document is focused primarily on characterizing risk to serve as an input for categorization, benefits should be considered as part of premarket authorization.

The term "SaMD" has evolved to include a more diverse landscape of software and varied interpretations across jurisdictions. The concepts presented in this document are not exclusive to any specific interpretation of the term SaMD. Rather the concepts can be helpful to consider more broadly for any software that meets the definition of a medical device or is part of a medical device.

In this document we refer to this relevant set of software as "medical device software" as a shorthand for document useability. This complex collection of medical device software includes various intersecting and distinct subsets, for example medical device software that:

- is intended to generate information for use in achieving one or more medical purpose;
- is part of a hardware medical device;
- is not part of a hardware medical device and is independent of other medical devices;
- is necessary for a hardware medical device to achieve its intended use/intended purpose;
- is driven or influenced by another medical device;
- has an output intended for a human user, medical device, and/or non-medical device; and
- uses inputs from humans, medical devices, and/or products that are not medical devices.

Medical device software can operate in complex socio-technical environments consisting of:

- software,
- hardware,
- information technology networks, and
- people

which form a complex and dynamic interaction between the software function, its inputs and outputs, the intended user, and the unique healthcare circumstances in which the software is used. This complexity together with:

- the interconnectedness of systems,
- the need for cybersecurity,
- the speed and frequency of development cycles,
- the speed at which a solution can be scaled up, and
- the various aspects of change implementation

contribute to the accurate depiction of a device and/or its risk profile. Medical device software can pose risks that are distinct and unique, such as those that relate to the information that is generated and output by the device and the capacity for varied degrees of autonomy. These devices may be used independently or as part of a platform and span a wide spectrum of risk profiles depending on the intended use, and potential harms associated with use and/or inaccurate outputs.

The clear and accurate characterization of medical device software is fundamental and supports device quality, risk management, regulatory decision-making, and device use in healthcare across the total product lifecycle. Stakeholders (including manufacturers, regulators, healthcare providers, end-users, and patients), to differing extents, will need to understand what medical device software is, its purpose, its context of use, how it works, and how it changes due to updates. This information can be necessary for proper use and to identify and evaluate the associated hazards, direct and indirect harms, risks and benefits, and to determine device risk classifications. As medical device software may change over its service life, it is important to revisit activities including medical device software and risk characterization as the software is updated or its scope or documentation changes.

Risk-based device classifications, applied in accordance with each jurisdiction's regulations, assign the appropriate regulatory obligations in each jurisdiction. Assigning risk categories to these devices can be challenging due to the broad range of technologies and characteristics that can influence risk, the variety of terminology and interpretations used to describe and qualify these devices, as well as the range of classification systems across global regulatory jurisdictions. This document identifies common considerations related to device characterization and risk characterization. Its aim is to offer a unified perspective and standardized language, thereby enhancing and fostering transparency and consistency between stakeholders. This document may help support comprehensive descriptions of medical device software, as well as thorough risk assessments for those devices.

Importantly, the considerations in this document are not intended to be used by stakeholders as a checklist or prescriptive means of medical device software characterization or determining software-specific device risks, or as an interpretation of any jurisdiction's laws and regulations. It is acknowledged that not all elements of this document will be applicable and pertinent to every medical device software.

2. Purpose and Scope

2.1. Purpose of the document

The purpose of this document is to promote and inform clear and accurate characterizations of medical device software, including developing an intended use/intended purpose statement. Additionally, it aims to introduce a general strategy for characterizing software-specific risks, drawing upon the essential components of a comprehensive characterization of medical device software.

This document is intended to:

- highlight the importance of comprehensive characterizations for medical device software to inform additional lifecycle activities including risk assessment and device categorization/classification;
- establish key features of and common vocabulary for the characterization of medical device software;
- identify fundamental elements of an intended use/intended purpose statement for medical device software;
- establish links between characterization features and risk for medical device software; and
- provide information for consideration during the identification and assessment of software-specific risks of medical device software.

2.2. Scope of the document

This document applies to the subset of software that meets the definition of a medical device (referred to throughout as *medical device software*), including software that meets the definition of Software as a Medical Device (SaMD) as is defined in the document, *IMDRF SaMD WG N10 Software as a Medical Device: Key Definitions*.

- This document is not intended to replace *IMDRF SaMD WG N12 "Software as a Medical Device": Possible Framework for Risk Categorization and Corresponding Considerations*. Rather, this document supplements and elaborates on the general SaMD characterization framework articulated in N12.¹ Importantly, the scope of this document is broader than that of N12, as this document applies to medical device software whereas the scope of N12 applies to the subset of devices referred to as SaMD. SaMD was defined in N12 as “software intended to be used for one or more medical purposes that perform these purposes without being part of a hardware medical device”. This document focuses on medical device software irrespective of the software technology and/or the platform (e.g., mobile application, cloud, server, hardware medical device).

¹ In particular, this document supplements and elaborates on the concepts in Section 5 of N12 (“Section 5 – Factors Important for SaMD Characterization”).

- This document is not intended to provide guidance on the classification of devices. Accordingly, this document does not provide guidance on the regulatory status of products. This document is not intended to be an interpretation of any jurisdiction's laws and regulations.
- This document focuses on software-specific risk considerations and is not intended to be comprehensive of all relevant risk considerations for a medical device software, which may also include additional risks related to interoperable or associated hardware, or breaches of data privacy. Where relevant, risk assessment should be carried out in consideration of the entirety of the device and not software-specific risk alone.
- This document is not intended to replace or conflict with existing risk management practices or the development of technical or process standards related to software risk management activities. This document is not a risk management document, rather it relies on established risk management principles, such as those in *ISO 14971 Risk Management for Medical Devices*, in the context of medical device software.
- This document is not intended to replace or conflict with existing IMDRF publications such as those published by the Artificial Intelligence (AI) or Cybersecurity Working Groups; however, it is acknowledged that there are direct relationships and overlap with those publications, and this document is intended to be complementary.
- The content in this document is not regulation or guidance and does not imply a convergence of regulations or categorization rules across jurisdictions. However, this document aims to establish harmonized concepts, considerations, and common vocabulary for the risk characterization of medical device software. Additional work may be required to apply and align these concepts in a given jurisdiction.

3. References

- *IMDRF/SaMD WG/N12 FINAL:2014 "Software as a Medical Device": Possible Framework for Risk Categorization and Corresponding Considerations*
- *IMDRF SaMD WG N10 FINAL:2013 Software as a Medical Device: Key Definitions*
- *IMDRF/GRRP WG/N52:2019 Principles of Labelling for Medical Devices and IVD Medical Devices*
- *GHTF/SG1/N77:2012 Principles of Medical Devices Classification*
- *ISO 14971:2019 Medical Devices - Application of Risk Management to Medical Devices*
- *AAMI TIR57: 2016/(R)2023 Principles for medical device security—Risk management*
- *ANSI/AAMI SW96:2023; Standard For Medical Device Security – Security Risk Management For Device Manufacturers*
- *IEC 80001-1:2021 Application of risk management for IT-networks incorporating medical devices*
- *IEC 80002-1:2009 Medical device software — Part 1: Guidance on the application of ISO 14971 to medical device software*
- *IEC 81001-5-1:2021 Health Software and Health IT Systems Safety, Effectiveness and Security – Part 5-1: Security – Activities in the product life cycle*
- *IEC 82304-1:2016 - Health Software — Part 1: General requirements for product safety*
- *IEC 62304:2006+AMD1:2015 Medical device software — Software life cycle processes*
- *AAMI TIR34971:2023 Application of ISO 14971 To Machine Learning In Artificial Intelligence—Guide*

4. Medical Device Software Characterization Considerations

The communication of a comprehensive medical device characterization (including the intended use/intended purpose and device description) supports stakeholders' ability to understand the device and characterize the associated risks and benefits. This will inform decision-making and help ensure device safety, effectiveness and proper use.

The characterization of medical device software, in some cases, constitutes a part of the device's description (e.g., for software within a hardware medical device that is needed to realize the intended use/intended purpose), while in other cases the characterization of the medical device software encompasses the entirety of the device's description. A comprehensive medical device software characterization is shaped by numerous elements, such as the medical purposes, intended users, intended use environment, and intended target populations, as well as the role and timing of the software's use and output in the clinical or healthcare workflow. The characterization clearly describes what the device is and what it is intended to do, as well as how, where, when and by whom the software is intended to be used and modified/changed.

This characterization information is essential for:

- identifying and validating the relevant user and clinical requirements,
- assessing the adequacy of supporting evidence,
- identifying and controlling risks,
- evaluating benefits,
- determining labelling and transparency requirements,
- managing medical device changes, and
- ensuring proper use while mitigating against misuse.

An accurate characterization of medical device software is also useful because it supports a complete view and clear understanding of risks associated with the device.

Sections 4.1 and 4.2 discuss considerations for manufacturers when characterizing medical device software within the intended use/intended purpose statement and device description. These considerations can support the determination of the pertinent and meaningful information to include within medical device software documentation, regulatory submissions, device labelling, and user interfaces. All features and attributes listed may not be relevant for every device but are included for consideration. What is communicated will be dependent on the stakeholder, regulatory jurisdiction, regulations, and the characteristics determined to have an impact on risk for the specific device.

4.1. Intended Use/Intended Purpose Statement

The intended use/intended purpose is defined within the *GHTF/SG1/N77 Principles of Medical Devices Classification* document as *the objective intent of the manufacturer regarding the use of a product, process or service as reflected in the specifications, instructions and information provided by the manufacturer.*

The concept of an intended use/intended purpose statement is familiar in many jurisdictions and is meant to capture the intended device function and medical purpose, including the indicated diseases, conditions, and/or circumstances for which the device is intended to be used. Such statements are generally most useful when they are sufficiently specific and avoid excessively general and/or open-ended language. It is acknowledged that the intended device function and indicated diseases may be considered separate in certain jurisdictions. However, for the purposes of this document, both are relevant and are suggested to be clearly described.

In order to foster and encourage clear and comprehensive intended use statements for medical device software, Key Elements of an intended use/intended purpose statement are captured in section 4.1.1 below. A sample statement guide can be found in Appendix A. It is important to note that not all elements will be applicable to every medical device software and the information provided in these sections is solely for consideration by manufacturers in the development of the medical device software labelling, documentation, and regulatory submissions, as appropriate. The sample statement may not be appropriate for all medical device software depending on the technology and intended use.

4.1.1. Key Elements of Intended Use/Intended Purpose Statement

1. Medical Purposes
2. Intended Disease or Condition
3. Intended Patient Populations
4. Intended Users
5. Intended Use Environment
6. Contraindications
7. Medical device software function, including:
 - Medical device software inputs
 - Medical device software outputs
 - Explanation of how the medical device software inputs and outputs fit into the clinical or healthcare workflow

4.2. Medical Device Software Description

A detailed medical device software description, accompanying the intended use/intended purpose statement, is often needed to ensure the comprehensive and adequate communication of all necessary characteristics and information related to a medical device software.

The following four subsections discuss detailed and interrelated information that can be relevant when characterizing a medical device software. They are organized according to the following four types or categories of information:

- Medical problem and/or objective
- Context of use
- Function and/or use
- Change management

The information within each category is presented in the form of characterization features with attributes. This non-exhaustive set of considerations for manufacturers is intended to highlight and clarify some important aspects when characterizing a medical device software. The features and attributes within each subsection are tabulated proceeding the discussion; the full set of features and attributes are provided in a summary table in Appendix B. These medical device software characterization considerations are also represented in Figure 1.

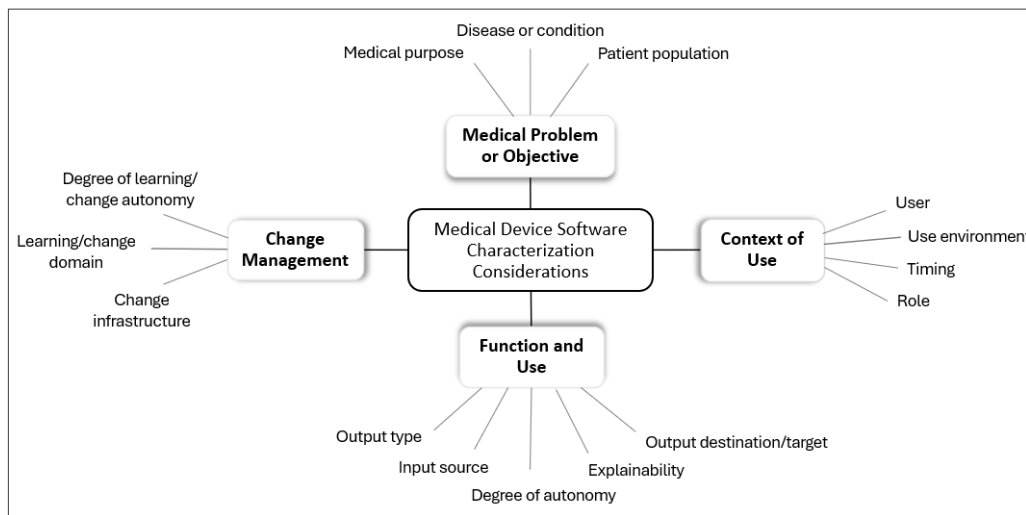


Figure 1 Medical device software characterization considerations

4.2.1. Medical Problem and/or Objective

The medical problem and/or objective addressed by a given medical device software is an important piece of the overall device characterization. This feature can be further broken down into the specific medical purpose, the intended conditions, and the intended patient population.

A medical device may be used in different stages of the care pathway, such as diagnosis (e.g., primary diagnosis, screening, triage, staging, etc.); treatment (e.g., relieving symptoms or restoring function); prevention (e.g., averting the occurrence of a disease or condition); prediction (e.g., disease prognosis), or monitoring (e.g., ongoing assessment of patient parameters). Understanding the specific medical purpose that the device performs or is used in achieving is a key part of characterizing the medical device software.

The condition or disease for which the medical device software is meant to be applied, and the general state of that condition or disease (for example, critical, serious or non-serious), are important pieces of information at the centre of characterizing a medical device software and determining the associated criticality or seriousness of the condition and importance of the output.

Finally, the intended patient population provides an important boundary within which the medical device software is meant to be used and is another defining feature of the medical device software characterization. In this document, the term patient is used to refer to individuals that receive or await healthcare with the use of the medical device software. The intended patients may be in a specific subgroup of the population (e.g., specific age, sex, gender, ethnicity, race, disability, diagnosis; or a fragile and/or vulnerable group; etc.), or specific intersection of subgroups of the population (e.g., specific age group + specific sex + those at risk of a specific condition).

The following table summarizes the identified features and attributes that help characterize the medical problem and/or objective. Please note that the content in this table is also summarized in Appendix B.

Table 1 Features and attributes for the characterization of the medical problem and/or objective

Characterization Feature	Potential Feature Attributes
Medical Purpose	Diagnosis (e.g., primary diagnosis, screening, triage, etc.), Prevention, Monitoring, Mitigation, Prediction, Treatment , etc.
Intended Disease or Condition	Critical, Serious, Non-Serious condition or disease , including consideration of the state of that condition (e.g., a chronic condition or an acute change in a chronic condition)
Intended Patient Population	<p>General population</p> <p>Specific subgroup of the population (e.g., fragile and/or vulnerable subgroup; specific age group, sex, gender, ethnicity, race, disability, diagnosis, etc.)</p> <p>Specific intersection of subgroups of the population (e.g., specific age group + specific sex + those at risk of a specific condition)</p>

4.2.2. Context of Medical Device Software Use

The characterization of medical device software extends beyond the device into the intended circumstances and setting for medical device software use. Two otherwise identical products with different intended contexts of use are distinct devices with different medical device software characterizations. Aspects of that context of use include the intended user of the medical device software, the intended use environment, timing within the healthcare task/intervention, and the role of the software and output within the healthcare task/intervention.

The intended user could be a non-clinical user, a non-physician medical professional, a general practitioner medical doctor (MD), a specialist physician, or a combination of these users with varying responsibilities in medical decision making. A non-clinical user, or lay-user, includes those users that are not trained or qualified to provide medical care, which might include a caregiver or patient user, or other users without medical qualifications such as community workers and health volunteers. Licensed medical professionals that are non-physicians include nurses, dentists, psychologists, radiation therapists, physiotherapists, etc. General practitioner (GP) medical doctors include, for example, primary care physicians or family doctors, while specialist physicians include radiologists, oncologists, dermatologists, psychiatrists, pathologists, surgeons, etc.

The intended use environment describes the setting in which patient care with the medical device software is meant to take place. This could be a non-clinical environment, a general healthcare environment, or a specialty healthcare environment. A non-clinical environment would include home-use or other settings outside of a clinical environment. General healthcare environments would include primary care clinics, dental offices, etc. A specialty healthcare environment would include, for example, emergency rooms, intensive care units, dermatology clinics, surgical operating rooms, and oncology departments within a hospital. When applicable, it can be important to specify the remoteness of the use environment (e.g., rural or isolated environments).

Another important aspect of the context of use is the finality of the software output and/or its significance in relation to the outcome of the healthcare task/intervention. The timing within the healthcare task/intervention is a feature that helps to contextualize the output in terms of being early, midway, or late in the healthcare task/intervention. Similarly, the role of the software and output within the healthcare task/intervention illustrates the relationship of the output amongst the steps in the healthcare task/intervention, in terms of relative chronology and the software's dependence on and/or input to the other steps. Taken together, these two features help to describe the impact or influence a medical device software may have on the overall trajectory and outcome of a patient's care. These features are important to understand the significance of the software's intended use and can help to identify where and how effects from the software's use can alter the course of a patient's healthcare experience.

The following table summarizes the identified features and attributes that can help characterize the context of use. Please note that the content in this table is also summarized in Appendix B.

Table 2 Features and attributes for the characterization of medical device software context of use

Characterization Feature	Potential Feature Attributes
Intended User	<p>Lay user/nonclinical user (e.g., caregiver, patient user, user without medical qualifications)</p> <p>Licensed medical professional, non-physician (e.g., registered nurse, dentist, psychologist, radiation therapist, physiotherapist, etc.)</p> <p>General Practitioner (e.g., primary care physician, family doctor, registered nurse practitioner)</p> <p>Specialist Healthcare Physician (e.g., radiologist, oncologist, dermatologist, pathologist, surgeon, etc.)</p>
Intended Use Environment	<p>Non-clinical Environment (e.g., home-use)</p> <p>General Healthcare Environment (e.g., primary care clinic, virtual primary healthcare)</p> <p>Specialty Healthcare Environment (e.g., hospital, specialty clinic, virtual specialty healthcare)</p>
Timing Within Healthcare Task/Intervention	<p>Early (e.g., triage, prediction of future diagnoses, early investigations upon suspicious symptoms or information, physiological signal or medical image acquisition for use in diagnosis or treatment planning)</p> <p>Midway (e.g., signal or image segmentation for use in diagnosis or treatment planning)</p> <p>Late (e.g., optimal image-guided treatment plan or dosage for consideration; adjunct diagnostic recommendations or second checks; continuous glucose monitor output analysis automatically driving basal insulin dosage; image-guided instrument control in robotic surgery)</p> <p>* Note: these 3 phases (Early, Midway and Late) described above serve as reference points, and it is not crucial to state which phase should be applied. Rather, it is important to characterize the timing of the output relative to the final intervention, decision, or action as well as the relative chronology of how the product will be introduced in relation to other steps (e.g., prior steps, concurrent steps, conditional steps, subsequent steps) and current standard medical practices.</p>
Role of Software and Output Within the Healthcare Task/Intervention	<p>Software and software output’s relationship to the healthcare task/intervention steps, such as the output’s <u>contribution</u> to the relevant healthcare decision or action (for example, intended as an aid that is combined with current practice); <u>alteration</u> of standard/current practice (for example, intended to replace or substitute all or part of current practice, to provide a new scheme, etc.); <u>dependence</u> on other steps (e.g., uses output values or clinical decisions from prior steps, concurrent steps, conditional steps); and/or <u>influence</u> over other steps (e.g., provides input to concurrent steps, subsequent steps, conditional steps, or final intervention/decision).</p>

4.2.3. Medical Device Software Function and/or Use

The function and use of a medical device software can be described by various aspects, such as its inputs, the generation of outputs, the output itself, and how that output fits into the care pathway. A data flow diagram and architecture diagram can also help support the communication of these elements.

The types of output provided by a medical device software could be a clinical interpretation or intervention, a workflow recommendation, or data or information for use in a medical purpose. Clinical interpretations or interventions can include, for example, a probability, prediction, detection, diagnosis, severity, prognosis, grade, or stage of a disease or condition; or the prescription, treatment, therapy, recommended dosage, or treatment plan for a disease or condition. A workflow recommendation, in contrast, is not an interpretation on the clinical decision or action but rather an intermediate step in the workflow, such as recommended contrast dye dosage; imaging technique, modality, or parameters; surgical tool choice; supplementary medical tests, etc. Data for use in medical purpose is output by a medical device software for use in a medical purpose and is typically more objective, such as anatomy measurements or volumes, segmented or contoured organs, tissues; processed, reconstructed, or de-noised images; processed signals or waveforms such as from electrocardiographs or electroencephalographs.

The input to the medical device software influences the function of the device and is fundamental to understanding the medical device software, the output, and the associated risks and considerations. The source of those inputs may be a human user (e.g., patient inputted symptoms or conversations), a medical device (e.g., a medical image), a non-medical device (e.g., data from a patient chart, medical record or electronic health record) or consumer product (e.g., smartphone photos, fitness tracker data). Notably, the inputs to a medical device software do not necessarily have to be medical information or come from a medical device. Regulators may consider the impact that non-medical data or data sources have on the safety and effectiveness of a medical device software. However, the use of non-medical data sources in a medical device software does not change the regulatory status of the source of non-medical data.

The degree of autonomy is a spectrum of capacities or liberties to operate independently of a user's direction, intervention and oversight. An autonomous device or medical device software provides outputs that impact the subsequent clinical action or decision without any user intervention. Conditionally autonomous outputs will meet this condition selectively (for example, for certain results, input characteristics, or circumstances). Supervised outputs can impact subsequent clinical actions or decisions without a user having to approve the output but operate under the supervision of users. Non-autonomous outputs are typically intended to help a user during their determination of a decision or action. The degree of clinical autonomy can be made clear by describing this aspect for clinical users specifically.

The level of explainability of the underlying logic is also an important characteristic of a medical device software. This can include information about the software algorithm or technology utilized (such as, deterministic formulae; machine learning approaches; mathematical simulations; etc.), relevant characteristics of the datasets used in development, and information about how an output or result was reached or the basis for a decision or action. This aspect could be characterized as (i) logic and output are explained and evaluable (e.g., a decision tree flow-chart structure showing decisions based on input features); (ii) logic and output are partially explained or partially evaluable (e.g., output provided with saliency maps); or (iii) logic and output are not explained or are unevaluable (e.g., Black Box, logic is considered too complex to be understood by intended audience). The level of information provided may depend on the intended audience of this information and their expected level of comprehension. The level of explainability of a medical device software contributes to the assessment of risks and uncertainties, as well as supporting evidence expectations.

The destination or target of the output could include human users, or medical devices (either with or without intermediate use by a human user).

The following table summarizes the identified features and attributes that can help characterize the device function and/or use. Please note that the content in this table is also summarized in Appendix B.

Table 3 Features and attributes for the characterization of medical device software function and/or use

Characterization Feature	Potential Feature Attributes
Output Type	<p>Clinical Interpretation or Intervention (e.g., diagnosis, suspicion, probability, prediction, detection, severity, prognosis, grade, stage, direct markers of a diagnosis, prescription, treatment/therapy, recommended treatment, recommended dosage, radiation treatment plan)</p> <p>Workflow Recommendation (e.g., contrast dye dosage; recommended imaging technique/modality/parameters; recommended surgical tool choice; recommended additional test based on established guidelines)</p> <p>Data for use in medical purpose (e.g., anatomy measurement, volume, or segmentation; processed image/image reconstruction/de-noised image; processed signal/waveform (e.g., processed ECG))</p>
Input Source	From human user, medical device, non-medical device or consumer product
Degree of Autonomy	<p>Independent/autonomous (i.e., output impacts subsequent clinical action or decision without user intervention)</p> <p>Conditionally independent/autonomous (output selectively impacts subsequent clinical action or decision without user in the loop. For e.g., software that independently filters normal findings but triggers clinician review of abnormal findings)</p> <p>Supervised (i.e., output impacts subsequent clinical action or decision without user having to approve, but with supervision from user who can intervene or stop the device)</p>

Non-autonomous (output considered by user in their determination of a decision/action)

***Note:** The Degree of Autonomy can be described with these terms or by leveraging terminology found in standards or other sources, as appropriate, provided the necessary information on the degree of autonomy and clinical autonomy are communicated.

<p>Explainability of Software and Output (underlying logic including the algorithm/technology used, relevant development data characteristics, and how an output is reached)</p>	<p>Logic and output are not explained or are unevaluable (e.g., Black Box, logic is considered too complex to be understood by intended audience)</p> <p>Logic and output are partially explained or can be partially evaluated (e.g., output provided with saliency maps)</p> <p>Logic and output are explained and can be evaluated (e.g., a decision tree flow-chart structure showing decisions based on input features accompanying training dataset characteristics)</p>
<p>Destination/Target of Output</p>	<p>Input to human user; Input to medical device</p>

4.2.4. Medical Device Software Change Management

How a device is intended to change is also part of the device characterization. This can include the degree of change implementation autonomy, the domain changes will apply to, and the supporting infrastructure such as installation locations and distribution channels.

The degree of learning or change management autonomy describes the degree of human oversight in the effectuation and control of training, learning and updates to the medical device software. Possible attributes within this feature can include self-learning (autonomous data collection, re-training and/or updates effectuated and controlled from within medical device software) and externally controlled learning (non-autonomous updates effectuated and controlled by the manufacturer and/or user).

The domain of learning or change implementation refers to the scope or applicable extent of change. This might be described as being applicable on a scale that is international, national, regional, clinic-specific, or patient-specific. For example, an update to a medical device software could be a tuned model based on a particular patient’s data that is intended for the patient alone, while another medical device software update could be a global model update intended to apply to all installations or instances of the software in a given country.

Another aspect of software change management is the infrastructure for installation, updates, and error corrections. Updates and changes to the software can be provided in response to software failures, cybersecurity vulnerabilities, errors, opportunities for improvement, critical performance updates, and recalls. Software-specific risks and risk controls can depend on the software distribution channels (app stores, manufacturer homepage, web application, etc.) and software installation locations (mobile phones and tablets, wearable technology, hardware medical devices, cloud, or personal computers (PCs) of the users, server anywhere in the world or one single server at the manufacturer site).

Distribution channels, such as app stores offering medical device software, may not be regulated in all jurisdictions and may have varying levels of control by the medical device manufacturer. Surveillance challenges and unclear responsibilities may occur in cases of recalls, field safety corrective actions and distribution of information. Furthermore, software installation location can influence the effectiveness and speed of access to updates or the deactivation of erroneous or recalled software and the traceability of affected installations and users.

The following table summarizes the identified features and attributes that can help characterize the device change management, including the degree of change autonomy, the change domain and infrastructure for installation, updates, and error correction. Please note that the content in this table is also summarized in Appendix B.

Table 4 Features and attributes for the characterization of medical device software change management

Characterization Feature	Potential Feature Attributes
Degree of Learning/Change Management Autonomy	<p>Self-learning/autonomous changing (autonomous updates effectuated and controlled from within medical device software)</p> <p>Externally controlled user-driven learning/change (non-autonomous updates effectuated and controlled by the user)</p> <p>Externally controlled manufacturer-driven learning/change (non-autonomous updates effectuated and controlled by the manufacturer)</p>
Domain of Learning/Change Implementation	International, National, Regional, Clinic/Site-specific, Patient-specific
Installation, Update and Error Correction Infrastructure	<p>Distribution channels (e.g., app stores, manufacturer homepage, web application)</p> <p>Installation locations (Mobile phones and tablets, wearable technology, hardware medical devices, cloud, or PCs of the users, server anywhere in the world or one single server at the manufacturer site)</p>

5. Medical Device Software Risk Characterization

Identifying and estimating medical device software-specific risk can raise unique questions compared to other medical devices. Risk management approaches, such as those proposed within ISO 14971, often describe risk as the combination of the probability of occurrence of harm and the severity of harm. Harms, however, can be both direct and indirect, and a comprehensive identification of software-specific contributions to possible harms can be challenging because software, *on its own*, may not pose “physical” hazards to which harms can be easily attributed. It is also important to consider when software can cause physical harm such as when autonomous devices provide outputs that impact the subsequent clinical action or decision without any user’s intervention; for instance, software that automatically controls the delivery of a substance to a patient.

Evaluating software-specific contributions to possible harms may require interpretation of primarily performance-related hazards², or, more specifically, information-related hazards, and understanding the associated risk is then critically tied to a complete understanding of a device’s intended use/intended purpose and particular implementation. In other words, when assessing the risk of medical device software, it is important to understand the contribution of information-related hazardous situations, which are closely tied to the role of software functionality in achieving an intended medical purpose. These hazardous situations can generally be understood through the lens of “performance-related hazards,” as described in ISO 14971, such as hazards relating to data access, availability, integrity, delivery, and diagnostic information as opposed to, for example, energy, biological, or chemical hazards.

An accurate characterization of medical device software, including its characteristics such as intended use, output type, use environment, autonomy, etc., allows for both a more comprehensive identification of these direct and indirect harms and a clear understanding of how software-specific harms can then lead to risks unique to a given intended use/intended purpose.

While the performance-related hazards and risks related to software do not always account for the *totality* of risk posed by a device (such as in the case of software that may supply data or generate the inputs for a hardware actuator that poses associated physical hazards), it is important to fully characterize the impact of a particular software implementation or solution on device risk because it can still lead to demonstrable impacts on patient safety or device effectiveness through direct or downstream means.

² ISO 14971:2019 Medical Devices – Application of Risk Management to Medical Devices

Further, it is important to consider that software-specific hazards often sit at the junction of both safety and cybersecurity risks. Therefore, it can be helpful to consider software-specific considerations pertaining to harm as a combination of how harm is defined for safety and cybersecurity. In other words, medical device software-specific consideration of harm could be viewed as relating to injury or damage to the health of people³ and reduction of effectiveness⁴ – where “reduction of effectiveness” can result from inadequate, incorrect, or absent data supplied to a human or product at an inappropriate time, rate, or with an inadequate method. For example, injection of unwanted or unintended bias into a decision-making system, whether or not it results in direct harm to a patient, can be understood as a harmful reduction in effectiveness. In other words, the introduction of the particular software solution has had a negative impact on the decision-making system. Often, this can also be viewed as accounting for “indirect harm” from the software, with potential direct impact to the patient as noted above.

Performance-related hazards pertinent to software – that is, specifically information-related hazards – can impact the function of other products or systems, how workflows or processes are informed, and can directly impact user decision making (e.g., software that outputs the wrong diagnostic information at an unacceptable rate). As such, a harmonized discussion on these topics can help promote a consistent and detailed understanding of these devices and how they may impact risk categorization across regulatory jurisdictions.

Key Points:

- When evaluating the risk posed by software (information and/or physical-based risks), both direct and indirect harms are important to consider.
- When hazards associated with software are information-based hazards (such as delayed, inappropriate, or erroneous information), it is important to consider potential harm as both injury or damage to health as well as a reduction in effectiveness when accounting for indirect harms.
- The possible harms and associated risks related to implementing software are dependent on a device’s specific intended use.

Below are general considerations for identification and analysis of software-specific hazardous situations, as well as considerations when including these hazardous situations as part of risk estimation. These approaches are intended to provide a shared means of discussing the unique risks posed by medical device software, and how such an understanding may drive device risk categorization across any number of risk categorization systems. The process for identification and analysis of these risks should be considered iteratively and should be carried out over the total product lifecycle of the device.

³ While ISO 14971:2019 defines harm as “injury or damage to the health of people, or damage to property or the environment” it can be helpful to consider, more specifically, harm as it relates to “injury or damage to the health of people” when discussing medical device safety in this document. The narrower definition of patient harm has the net effect of prioritizing regulatory review of those changes necessary to protect public health.

⁴ Harm is defined in TIR57: 2016/(R)2023 as “physical injury or damage to the health of people, or damage to property or the environment, or reduction in effectiveness, or breach of data and systems security” as described in IEC 80001-1:2021.

5.1. Identification and Analysis

The success of risk assessment and management activities hinges on the risk assessors' understanding of what the medical device software is and is meant to do, as well as how, where, when and by whom the medical device software is meant to be used. The comprehensive characterization of medical device software, considering the information presented in section 4 of this document, provides the foundation necessary for software-specific risk characterization. Approaches to identifying and considering risks within each of the information groupings in section 4 are provided below, in part, to illustrate the way many variables contribute and interact to form a more complete understanding of the unique risks that may impact a particular medical device software.

To identify and characterize software risk, including relevant risks related to cybersecurity, it is helpful to step through the process of first identifying a device characteristic, then asking *why* the characteristic matters to the intended use/intended purpose of the software, and then identifying the hazardous situations that may arise based upon both the intentional software design decisions and unintentional software failures. It remains important, however, to ensure that exploring device characteristics in this manner is not done in a vacuum and interdependencies of the software are carefully considered to comprehensively describe a medical device software's "risk characterization."

Appendix C provides questions for consideration to accompany each characterization feature previously identified in section 4. These questions are provided to help develop an understanding of "why the characteristic matters to the intended use/intended purpose of the software," as a means to help identify specific hazardous situations that may be related to the software's design and intended use/intended purpose. While not comprehensive, the questions aim to highlight how the context provided by each of a device's unique characteristics could impact an understanding of the potential harms introduced by a particular software, and thereby affect the overall risk of the medical device. The questions are intended to help guide a thorough consideration of potential harms a medical device software could introduce prior to the introduction of risk controls/ mitigations, and not all questions may be applicable or relevant to every medical device software.

Appendix D includes examples illustrating how answering the questions in Appendix C can help to identify the way different characterization features and their interactions may affect an understanding of the risks introduced by a particular medical device software. Importantly, identifying these "software-specific" contributions to device risk is intended as a means of articulating why the software for a particular medical use/purpose may or may not alter device risk categorization under any number of frameworks. This concept is discussed further in section 5.3 of this document.

5.2. Estimation

As noted above, risk management approaches, such as those proposed within ISO 14971, often describe risk as the combination of the probability of occurrence of harm and the severity of harm. These risk estimation features, together with medical device software characterization features outlined in section 4 of this document, can be essential in assessing and managing risks.

For medical device software, risk management requires the identification of the potential *direct and indirect* harms associated with hazardous situations, such as erroneous outputs from the software, followed by an assessment of the severity of those harms, such as reductions in life expectancy, psychological injury, or inappropriate or unnecessary invasive treatment and/or test. While probability of harm can generally be helpful to consider when estimating risk, there is not broad consensus on a method for quantitatively estimating probability of occurrence of software failure. Additionally, cybersecurity risk management often considers exploitability of vulnerabilities rather than probability of occurrence of harm; and it is generally understood that probability of software-related harms can be influenced by factors like usability, which can make estimation further challenging.⁵ To this end, when estimating software-specific risks, it can be helpful to set the probability of software failure to 1 and if possible, estimate the probability based on other factors to perform risk estimation.

The guiding questions in Appendix C can provide a basis for isolating software-specific hazards *and* for contextualizing their potential severity of harm, by understanding how applying a specific software solution can affect the way the medical device intended use/intended purpose is achieved.

These concepts can then be leveraged for risk characterization (e.g., through risk assessment per ISO 14971) and the determination of the severity of direct or indirect harm caused by a given, software-specific hazardous situation (e.g., catastrophic, critical, serious, minor, negligible). Once harms are identified, the approach to “software-contributed” risk estimation is not unique. That said, when software may need to be considered in the context of a broader device to achieve an intended use/intended purpose, it can also be helpful to consider whether the software becomes a single-point failure for a given possible harm and, if so, how this may impact risk estimation and associated mitigations.

5.3. Approaches for Risk Categorization

Software risk characterization is important to be performed early in the design phase of the life cycle. As with risk characterization discussed throughout this document, risk categorization should be performed prior to implementation of risk controls (rather than residual risk) for medical device software. It may not be universally possible or beneficial to create completely rigid and distinct categories of risk for any one type of function, disease, intervention, population, or user. For any given medical device software, there may be both interdependencies and unequal weight amongst characterization groupings that ultimately inform the understanding of device risk and, therefore, may impact a subsequent categorization. Further, when addressing the specific contribution to device risk posed by software, considerations like how supplied information will be ingested by a given userbase, as just one example, may reasonably not have uniform or universal answers across jurisdictions.

⁵ Ref: IEC 62304, AAMI TIR57, AAMI TIR34971

Different jurisdictional authorities may have distinct philosophies and legal obligations which shape their different risk-based classifications. Therefore, the discussion provided in section 5 and further illustrated in Appendix B is intended as a common basis for considering and articulating how characterization features impact the risk of software that meets the definition of a medical device, particularly through the lens of the interdependent factors shaping an understanding of risk specific to software for a given intended use/intended purpose. Put another way, this document intends to provide insight into *how* a particular software risk categorization could be reached without prescribing a single “correct” and universal category to any given device. As noted previously, among other complications, software-specific risks may have a significant but not exclusive influence on the risk categorization applicable to a given device. In premise, this document can serve as the basis for discussing a given understanding of a medical device software’s risk within a broader device system or regulatory structure.

6. Considerations for Implementation

Providing a common basis for describing medical device software and considering how different characteristics impact risk can help promote safety and effectiveness as well as consistency and alignment across jurisdictions.

The considerations presented in sections 4.0 and 5.0 can be used to support understanding of a medical device software and its risks and facilitate the interpretation and application of different device risk classification systems across jurisdictions. While the medical device software characterization described in this document relates to evaluating the risk of the device, it is important that any regulatory assessment also considers the benefit to health from the use of the device and weighs the benefits against any risks of the device.

Device classification in a given jurisdiction will ultimately be dictated by the governing authorities, laws, and regulations. To the extent possible, jurisdictions may consider incorporating harmonized language and concepts from this document into their local guidance or processes, for example, connecting the device and risk characterization language in the document to their labelling and risk management expectations or classification regulations.

Jurisdictions may be able to leverage a subset of characterization features and attributes, together with the assessment of medical device software risks and their severity, to describe their approach to applying risk categorization to medical device software.

These concepts are intended to be used by stakeholders alongside their existing frameworks, to provide additional detail and exposition for decision-making – ultimately promoting and informing clear, consistent, and accurate characterizations of medical device software.

Appendix A: Sample Intended Use/Intended Purpose Statement

In order to foster and encourage clear and comprehensive intended use statements for medical device software, Key Elements of an intended use/intended purpose statement are captured in section 4.1.1. A sample statement guide can be found below. It is important to note that not all elements will be applicable to every medical device software and the information provided in section 4.1.1 and below is solely for consideration by manufacturers in the development of the medical device software labelling, documentation, and regulatory submissions, as appropriate. The sample statement may not be appropriate for all medical device software depending on the technology and intended use. Although typically included in the intended use/intended purpose statement, for some devices, information such as contraindications may be included elsewhere in the medical device software labelling due to the volume of information.

The [name of medical device software] is software intended for use in the [medical purposes] of [conditions/diseases/disorders] in [intended patient populations]. This software is intended to be used by [intended user populations] in [intended use environments]. This medical device software is contraindicated for [contraindications]. This medical device software uses [inputs] in order to produce [description of outputs]. These outputs are [description of how the output is intended to be used, how it fits in the clinical or healthcare workflow and how it contributes to the final healthcare decision/action].

Appendix B: Characterization Feature Summary Table

The following table represents an aggregated version of the four tables presented separately in Section 4, Medical Device Software Description.

Information Grouping	Characterization Feature	Potential Feature Attributes
Medical Problem and/or Objective	Medical Purpose	Diagnosis (e.g., primary diagnosis, screening, triage, etc.), Prevention, Monitoring, Mitigation, Prediction, Treatment , etc.
	Intended Disease or Condition	Critical, Serious, Non-Serious condition or disease , including consideration of the state of that condition (e.g., a chronic condition or an acute change in a chronic condition)
	Intended Patient Population	General population Specific subgroup of the population (e.g., fragile and/or vulnerable subgroup; specific age group, sex, gender, skin tone, race, disability, diagnosis, etc.) Specific intersection of subgroups of the population (e.g., specific age group + specific sex + those at risk of a specific condition)
Context of Medical Device Software Use	Intended User	Lay user/nonclinical user (e.g., caregiver, patient user, user without medical qualifications) Licensed medical professional, non-physician (e.g., registered nurse, dentist, psychologist, radiation therapist, physiotherapist, etc.) General Practitioner (e.g., primary care physician, family doctor, registered nurse practitioner) Specialist Healthcare Physician (e.g., radiologist, oncologist, dermatologist, pathologist, surgeon, etc.)
	Intended Use Environment	Non-clinical Environment (e.g., home-use)

General Healthcare Environment (e.g., primary care clinic, virtual primary healthcare)

Specialty Healthcare Environment (e.g., hospital, specialty clinic, virtual specialty healthcare)

Timing Within Healthcare Task/Intervention

Early (e.g., triage, prediction of future diagnoses, early investigations upon suspicious symptoms or information, physiological signal or medical image acquisition for use in diagnosis or treatment planning)

Midway (e.g., signal or image segmentation for use in diagnosis or treatment planning)

Late (e.g., optimal image-guided treatment plan or dosage for consideration; adjunct diagnostic recommendations or second checks, continuous glucose monitor output analysis automatically driving basal insulin dosage; image-guided instrument control in robotic surgery)

*** Note:** these 3 phases (Early, Midway and Late) described above serve as reference points, and it is not crucial to state which phase should be applied. Rather, it is important to characterize the timing of the output relative to the final intervention, decision or action as well as the relative chronology of how the product will be introduced in relation to other steps (e.g., prior steps, concurrent steps, conditional steps, subsequent steps) and current standard medical practices.

Role of Software and Output Within the Healthcare Task/Intervention

Software and software output's relationship to the healthcare task/intervention steps, such as the output's contribution to the relevant healthcare decision or action (e.g., intended as an aid that is combined with current practice); alteration of standard/current practice (e.g., intended to replace or substitute all or part of current practice, to provide a new scheme, etc.); dependence on other steps (e.g., uses output values or clinical decisions from prior steps, concurrent steps, conditional steps); and/or influence over other steps (e.g., provides input to concurrent steps, subsequent steps, conditional steps, or final intervention/decision).

<p>Medical Device Software Function/ Use</p>	<p>Output Type</p>	<p>Clinical Interpretation or Intervention (e.g., diagnosis, suspicion, probability, prediction, detection, severity, prognosis, grade, stage, direct markers of a diagnosis, prescription, treatment/therapy, recommended treatment, recommended dosage, radiation treatment plan)</p> <p>Workflow Recommendation (e.g., contrast dye dosage; recommended imaging technique/modality/parameters; recommended surgical tool choice; recommended additional test based on established guidelines)</p> <p>Data for use in medical purpose (e.g., anatomy measurement, volume, or segmentation; processed image/image reconstruction/de-noised image; processed signal/waveform (e.g., processed ECG))</p>
	<p>Input Source</p>	<p>From human user, medical device, non-medical device or consumer product</p>
	<p>Degree of Autonomy</p>	<p>Independent/autonomous (i.e., output impacts subsequent clinical action or decision without user intervention)</p> <p>Conditionally independent/autonomous (output selectively impacts subsequent clinical action or decision without user in the loop, e.g., software that independently filters normal findings but triggers clinician review of abnormal findings)</p> <p>Supervised (i.e., output impacts subsequent clinical action or decision without user having to approve, but with supervision from user who can intervene or stop the device)</p> <p>Non-autonomous (output considered by user in their determination of a decision/action)</p>
	<p>Explainability of Software and Output (underlying logic including the algorithm/technology used, relevant development data characteristics, and how an output is reached)</p>	<p>*Note: The Degree of Autonomy can be described with the terms above or by leveraging terminology found in standards or other sources, as appropriate, provided the necessary information on the degree of autonomy and clinical autonomy are communicated.</p> <p>Logic and output are not explained or are unevaluable (e.g., Black Box, logic is considered too complex to be understood by intended audience)</p> <p>Logic and output are partially explained or can be partially evaluated (e.g., output provided with saliency maps)</p>

	Destination/Target of Output	<p>Logic and output are explained and can be evaluated (e.g., a decision tree flow-chart structure showing decisions based on input features accompanying training dataset characteristics)</p> <p>Input to human user; Input to medical device</p>
Medical Device Software Change Management	Degree of Learning/Change Management Autonomy	<p>Self-learning/autonomous changing (autonomous updates effectuated and controlled from within medical device software)</p> <p>Externally controlled user-driven learning/change (non-autonomous updates effectuated and controlled by the user)</p> <p>Externally controlled manufacturer-driven learning/change (non-autonomous updates effectuated and controlled by the manufacturer)</p>
	Domain of Learning/Change Implementation	<p>International, National, Regional, Clinic/Site-specific, Patient-specific</p>
	Installation, Update and Error Correction Infrastructure	<p>Distribution channels (e.g., app stores, manufacturer homepage, web application)</p> <p>Installation locations (mobile phones and tablets, wearable technology, hardware medical devices, cloud, or PCs of the users, server anywhere in the world or one single server at the manufacturer site)</p>

Appendix C: Example Considerations to Understand Software Hazards Associated with Device Design and Intended Use

The questions noted in the below table are intended to help guide a thorough consideration of potential harms that a medical device software could introduce. Not all questions may be applicable or relevant to every medical device software. This is not intended to be an exhaustive or required list of considerations for the intended use or the intended user of the medical device software, rather they are optional examples that may be helpful to consider while characterizing software risk.

Information Grouping	Characterization Feature	Considerations for Medical Device Software Risk Characterization
<p>Medical Problem and/ or Objective</p>	<p>Medical Purpose</p>	<ul style="list-style-type: none"> •Is the medical device software intended to be used as adjunctive or alongside other tools or treatment? Is the medical device software intended to replace or augment a system or process? If it is meant to augment, in what manner is the medical device software augmentative (for example, is the software output additive or confirmatory to another process or outcome)? •Is the output of the software itself intended to be therapeutic or a treatment? Is the software output used for decision-making with diagnostic or therapeutic purposes? Is the software used to monitor physiological processes or vital physiological parameters? Does the software have alarm functions used to prompt immediate or near-term intervention?
	<p>Intended Disease or Condition</p>	<ul style="list-style-type: none"> •How, if at all, does the condition/disease (for example, acute or chronic) that the medical device software is intended for impact the criticality of the data output by the software? •Does the condition/disease modify the timing of when the information is needed or is provided or must be used? •Does the condition/disease define the sensitivity or accuracy of the information needed for the input or output of the software? Could the nature of variation of monitored parameters result in immediate or near-term danger to the patient? •Could the decisions or diagnostics made by the software output have an impact that may cause death or an irreversible deterioration in condition/disease or a serious deterioration in condition/disease or a surgical intervention?
	<p>Intended Patient Population</p>	<ul style="list-style-type: none"> •Does the intended patient population include a specific vulnerable subgroup? •How diverse is the intended patient population? How generalized does the information need to be to perform adequately across the intended patient population? How specific? •Do the decisions or diagnostics provided by the software output reflect the intended user demographics?
<p>Context of Medical Device Software Use</p>	<p>Intended User</p>	<ul style="list-style-type: none"> •Does the medical device software enable new/different users to achieve the clinical task than those who would perform the task without the software? •Does the user need to possess expertise, or access to expertise (such as specific training to use the software), to understand the inputs and/or outputs of the software?
	<p>Intended Use Environment</p>	<ul style="list-style-type: none"> •Is use of the medical device software providing a clinical task or service in an environment that would not

Timing Within Healthcare Task/Intervention

- **otherwise have such tasks or services** available (e.g., would otherwise require an expert present)?
- Is the device intended to be used in an uncontrolled or unconventional setting?
- Can external factors, both physical (e.g., noise, lighting) and digital (e.g., network connectivity), affect the use, input or output of the device?
- Do the expected virtual conditions and computing environment require additional software controls (e.g., hardware or software compatibility verification or authentication) and/or impact users' access to the software?
- Does the user have **adequate time to review the basis** for the information output by the software or to review and curate the information being used as input to the software?
- Could the software output **initiate a healthcare intervention that would not otherwise be identified** by a particular user or in a particular setting (e.g., pre-screening information prompting a patient to speak to a doctor about a possible condition)?
- Are there **possible harms or dangers related to the healthcare task/intervention that could occur immediate or near-term** to the software's outputs?
- Are there **possible harms or dangers related to the healthcare task/intervention that could occur distantly** from the use of the software, but are related to decision points generated by the software's outputs?

Role of Software and Output Within the Healthcare Task/Intervention

- Does an erroneous output from the software at the intended point in the workflow **put the patient on a path toward subsequent harm**?
- Is the **frequency of output** appropriate to its role and timing in the workflow (e.g., is there a potential for notification fatigue, i.e., desensitization of the user to alarms, alerts, or notifications)?
- Does the software create a **single point of failure** in the clinical task/intervention?

Device Function/Use	Output Type	
		<ul style="list-style-type: none"> • Is the output supplementing additional information to contribute to a clinical interpretation or workflow recommendation? Is it a replacement or substitution for information meant to determine a clinical interpretation, workflow recommendation, or as data for use in a medical purpose? • Is the output commonly accepted in clinical practice or based upon sound scientific principles? Is the output proprietary? • Is access to the output tiered or limited by user or other credentials?

<p>Input Source</p>	<ul style="list-style-type: none"> •Is the output Boolean, e.g., values that are either true or false? •Is the input source from a human user, medical device, non-medical device or consumer product? •Is the input source unique or could the data be obtained through other methods or sources? •Is an adequate input source governed by specific parameters such as rate, sensitivity, or precision (inclusion and exclusion criteria)? Is the input relevant? •Is the input data direct or informed or transformed by other tools, products, or intermediaries? Are the transformed data suitable? •Are there multiple input sources or data types? Are they interdependent? •Does the data the software is processing accurately reflect the demographics, backgrounds, and characteristics of the population the software will be used for?
<p>Degree of Autonomy</p>	<ul style="list-style-type: none"> •Is a user in the loop? Is the user in the loop a health care professional?
<p>Explainability of Software and Output (underlying logic including the algorithm/ technology used, relevant development data characteristics, and how an output is reached</p>	<ul style="list-style-type: none"> •Is the relevant functionality of the product sufficiently explained and reasonably understood by the patient? •Is the relevant functionality of the product explained and understood by users other than the patient? Is different information provided to different user groups or patients? •Is the relevant functionality partially explained or partially able to be evaluated by the user (e.g., output provided with saliency maps)?
<p>Destination/Target of Output</p>	<ul style="list-style-type: none"> •Is the output the only instruction/data/information needed to drive the target's next action?
<p>Medical Device Software Change Management</p>	<ul style="list-style-type: none"> •Does the medical device software independently change its underlying algorithms? •How often is medical device software performance verified? •Are updates to algorithmic performance driven by non-clinical or clinical users, or manufacturer driven, or a combination of these users?
<p>Domain of Learning/Change Implementation</p>	<ul style="list-style-type: none"> • Is domain-specific implementation necessary to achieve adequate software performance? • Where are changes intended to be implemented and how variable are these domains?
<p>Installation, Update and Error Correction Infrastructure</p>	<ul style="list-style-type: none"> •What specific channels are used to distribute the medical device software?

- Does the medical device software have **multiple installation locations**? Where are corrections initiated?
 - How are updates pushed/deployed (e.g., automatically or manually, remotely or on site)?
-

Appendix D: Examples of Linking Characterization Features and Risk

This section includes examples applying the considerations described in sections 4 and 5. The examples below are intended to help illustrate how robustly characterizing software and systematically assessing the contribution of characterization factors to the software risk can provide a shared and more granular means of discussing risk that remains transferrable between potentially diverse risk categorization structures.

Below we have provided a full example of a software function applying the considerations discussed in the above sections as well as specific examples highlighting how changes in specific groupings of characterization features may impact risk. While the assessments below relate to characterizing the risk of the medical device, it is important that any regulatory assessment also considers the benefit to health from the use of the medical device and weighs the benefits against any risks of the medical device.

It is important to note when assessing the contribution of characterization factors to software risks that software products may have multiple functions, including both medical device and non-medical device functions. If a medical device has multiple functions, each function should be assessed separately in accordance with its respective regulatory status. If a software product has a non-medical device function, the assessment for that non-medical device function should only be related to the impact that the non-medical device function has on the medical device function.

Example A: Software function that serves as a primary diagnostic to identify patients with prediabetes

Scenario: *The software is intended to analyze health-related data including data from electronic health records, laboratory tests, and other diagnostic tests to identify individuals with pre-diabetes (i.e., an early marker of diabetes) with an output that is reviewed by healthcare practitioners.*

A.1 Sample Intended Use/Intended Purpose Statement

The **Product X** is software intended for use in the **diagnosis of prediabetes in adults at risk of developing diabetes**. This software is intended to be used by **medical professionals** in **general healthcare environments**. This medical device software is developed using **a machine learning model**. This medical device software is used for **patients without an existing diabetes diagnosis**. This medical device software uses **specific data within the electronic health records (EHR)** in order to produce a **conditionally automatic algorithm output that provides likelihood of developing diabetes**. These outputs are **conditionally independent/ autonomous** (i.e., output is presented to healthcare providers for review above a threshold %) and are intended to be used as a **clinical workflow recommendation for additional testing or follow-up based on established guidelines**.

As discussed in Appendix C, addressing each of the characterization features through corresponding questions is helpful for evaluating risk. The below questions are listed by information grouping to support comprehensive discussion of risk considerations.

A.2 Software Risk Considerations:

A.2.1 Medical Problem and/or Objective

Characterization Feature	Considerations for Medical Device Software Risk Characterization	Discussion
Medical Purpose	<ul style="list-style-type: none"> •Is the medical device software intended to be used as adjunctive or alongside other tools or treatment? Is the medical device software intended to replace or augment a system or process? If it is meant to augment, in what manner is the medical device software augmentative (for example, is the software output additive or confirmatory to another process or outcome)? •Is the output of the software itself intended to be therapeutic or a treatment? Is the software output used for decision making with diagnostic or therapeutic purposes? Is the software used to monitor physiological processes or vital physiological parameters? Does the software have alarm functions used to prompt immediate or near-term intervention? 	<p>In considering the Medical Purpose, this medical device software is intended to be used alongside other tools or treatments, i.e., used alongside additional diagnostic test results, treatments, and data available in electronic health records.</p> <p>The medical device software is intended to augment a system or process, i.e., the software output is used as a tool to aid in the diagnosis of pre-diabetes. Here, it is helpful to consider that the software is intended to augment and aid, which suggests the output may not be the sole influence on the related clinical decision point. If the software output is not a single point failure that will lead to patient harm, this can impact our understanding of the software’s risk.</p>

<p>Intended Disease or Condition</p>	<ul style="list-style-type: none"> •How, if at all, does the condition/disease (for example, acute or chronic) that the medical device software is intended for impact the criticality of the data output by the software? •Does the condition/disease modify the timing of when the information is needed or is provided or must be used? •Does the condition/disease define the sensitivity or accuracy of the information needed for the input or output of the software? Could the nature of variation of monitored parameters result in immediate or near-term danger to the patient? •Could the decisions or diagnostics made by the software output have an impact that may cause death or an irreversible deterioration of condition/disease or a serious deterioration in condition/disease or a surgical intervention? 	<p>When considering the Disease or Condition of the patient, it is helpful to consider that the general state of the condition as a pre-disease state (i.e., the state of a condition before it is a disease) does not impact the criticality of the output of the software. The general state of the condition being a pre-disease state determines that the information is needed or must be used before the disease (diabetes) is diagnosed to predict a high likelihood of subsequently developing the disease (diabetes). Furthermore, the general state of the condition as a pre-disease state and the likelihood of a pre-diabetes state being present (i.e., pre-test probability) determines the sensitivity and/or accuracy of the information needed for the output of the software. Given these factors, the software output is unlikely to have an impact that may cause death or an irreversible deterioration of condition/disease, which can be helpful to consider when evaluating the overall impact that a software failure could have on the device risk. In this case, the risk may be generally lower, because the output's relationship to the condition is not one that may likely lead to irreversible harm.</p>
<p>Intended Patient Population</p>	<ul style="list-style-type: none"> •Does the intended patient population include a specific vulnerable subgroup? •How diverse is the intended patient population? How generalized does the information need to be to perform adequately across the intended patient population? How specific? •Do the decisions or diagnostics provided by the software output reflect the intended user demographics? 	<p>The Intended Patient Population in which this medical device software is intended to be used includes the general public but may include vulnerable subgroups such as individuals of different ethnicities, or different age groups (e.g., <40, 40-60, >60 years old). The intended patient population is the general public that is representative of the demographics in the local userbase which may include regional, state, or national level. This information needs to be broadly generalizable to perform adequately. As a diagnostic aid the performance of the software must have adequate sensitivity and specificity; however, the performance is dependent on the prevalence of the condition (i.e., pre-diabetes) being tested. Because this software is intended for a general population, the software may need to operate in consideration of a wide variety of patients in the intended population.</p>

A.2.2 Context of Device Use

Characterization Feature	Considerations for Medical Device Software Risk Characterization	Discussion
Intended User	<ul style="list-style-type: none"> •Does the software enable new/different users to achieve the clinical task than those who would perform the task without the software? •Does the user have the expertise, or access to expertise (such as specific training to use the software), to understand the inputs and/or outputs of the software? 	<p>In considering the Intended User, this medical device software enables both new and different users (i.e., different Health Care Providers (HCPs)) to achieve the clinical task (i.e., to identify individuals with pre-diabetes) that would otherwise not be performed without the software. This medical device software can be used by different intended users (i.e., different primary care and/or specialty HCPs). The software is analyzing health-related data in electronic health records that does not require the user (i.e., HCP) to have specialized training. This medical device software requires the user (i.e., HCP) to have the necessary expertise to understand the input (i.e., type of data in electronic health records that the software analyzes) and the output (i.e., pre-diabetes) produced by the software.</p>
Intended Use Environment	<ul style="list-style-type: none"> •Is use of the medical device software providing a clinical task or services in an environment that would not otherwise have such tasks or services available (e.g., would otherwise require an expert present)? •Is the device intended to be used in an uncontrolled or unconventional setting? •Can external factors, both physical (e.g., noise, lighting) and digital (e.g., network connectivity), affect the use, input or output of the device? •Do the expected virtual conditions and computing environment require additional software compatibility verification or authentication) and/ or impact the users' access to the software? 	<p>The Intended Use Environment for this medical device software includes providing services in a healthcare (i.e., clinical) environment and is not intended to function outside healthcare settings or in those settings where healthcare is not being delivered with access to an electronic health record (i.e., settings using paper-based records). External factors (i.e., those factors that can impact the function of the medical device software) such as physical (e.g., physical related factors) and digital (e.g., broadband, internet connectivity, access issues to different healthcare databases) factors, may have a minor or negligible effect on the use, input, or output of the device. Further, the restricted intended use environment reduces the variability of operating conditions where the software must perform adequately.</p>

Timing Within Healthcare Task/Intervention

- Does the user have **adequate time to review the basis** for the information output by the software or to review and curate the information being used as input to the software?
- Could the software output **initiate a healthcare intervention that would not otherwise be identified** by a particular user or in a particular setting (e.g., pre-screening information prompting a patient to speak to a doctor about a possible condition)?
- Are there **possible harms or dangers related to the healthcare task/intervention that could occur immediate or near-term** to the software’s outputs?
- Are there **possible harms or dangers related to the healthcare task/intervention that could occur distantly** from the use of the software, but that are related to decision points generated by the software’s outputs?

As for the Timing within Healthcare Task/Intervention, the output of this medical device software is considered routine and non-urgent. The user has adequate time to review the output of this medical device software and to curate and review the basis or information used as its input. Because of the intended timing, the impact of the software’s risks may overall be considered lower than those risks might be in a time-critical or urgent use case. Because some patients for whom review might be impactful to their future care could be missed if the software does not present their cases for review, there is a possible harm that could occur distantly from the use of the software.

Role of Software and Output Within the Healthcare Task/Intervention

- Does an erroneous output from the software at the intended point in the workflow **put the patient on a path toward subsequent harm**?
- Is the **frequency of output** appropriate to its role and timing in the workflow (e.g., is there a potential for notification fatigue, i.e., desensitization of the user to alarms, alerts, or notifications)?
- Does the software create a **single point of failure** in the clinical task/intervention?

When considering the Role of Software and Output Within the Healthcare Task/Intervention, as a recommendation for further testing, the risk of output from the software at the intended point in the workflow putting the patient on a path toward subsequent harm is low. The frequency of output from the software and timing in the clinical workflow do not present risks of notification fatigue. The software also does not present a single point of failure in the clinical task/intervention as other data within the patient’s primary care routine is available to identify symptoms of prediabetes

When considering questions related to the Context of Device Use, the Intended User for the medical device software in the scenario provided is limited to healthcare practitioners in the Intended Use Environment of a health care facility. This, in combination with the Timing Within Healthcare Task/Intervention and Role of Software and Output Within the Healthcare Task/Intervention considerations indicates that these characterisation features pose a lower impact on overall risk characterization.

A.2.3 Device Function/Use

Characterization Feature	Considerations for Medical Device Software Risk Characterization	Discussion
--------------------------	--	------------

Output Type

- Is the output **supplementing** additional information to contribute to a clinical interpretation or workflow recommendation? Is it a **replacement or substitution** for information meant to determine a clinical interpretation, workflow recommendation, or as data for use in a medical purpose?
- Is the output commonly accepted in **clinical practice** or based upon sound **scientific principles**? Is the output **proprietary**?
- Is access to the output **tiered or limited** by user or other credentials?
- Is the output **Boolean**, e.g., values that are either true or false?

In considering the Output Type, this medical device software provides additional information (i.e., diagnosis of a pre-diabetes state) that supplements clinical recommendations (e.g., for subsequent diagnostic testing) with data that is used for a medical purpose (e.g., recommendations for lifestyle modification and/or treatments). The output of this medical device software is commonly accepted in clinical practice (i.e., the diagnosis of pre-diabetes) and, provided it has been adequately validated with an appropriate indication for use, is based on sound scientific principles. This medical device software output is considered proprietary because the specific calculation to arrive at a threshold to present the output to the HCP for review is devised by the company and is not simply a well-known and accepted threshold or calculation. Access to the output of this medical device software is first made available to the HCP who ordered the use of this software (i.e., analysis of health-related data for an individual who does not have pre-diabetes to determine if pre-diabetes is present in this individual). Thereafter, the output of this medical device software is accessible by HCPs who are providing care to this individual and information is not withheld from the HCP on the basis of a specific product access tier. The information is also not meant to be shared with a wide variety of users such that varying levels of access is implemented, such as might be the case if the product's outputs were meant for review by both patients and their providers.

Input Source

- Is the input source from a human user, medical device, non-medical device or consumer product?
- Is the input source **unique** or could the data be obtained through **other methods or sources**?
- Is an adequate input source **governed by specific parameters** such as rate, sensitivity, or precision (inclusion and exclusion criteria)? Is the input relevant?
- Is the input data direct or **informed or transformed** by other tools, products, or intermediaries? Are the transformed data suitable?
- Are there **multiple input sources** or data types? Are they **interdependent**?

The Input Source of this medical device software is unique and limited to the data that is available in electronic health records for individuals in whom this software will be used. The input data cannot be obtained through other methods or sources. The input source of this medical device software is governed by specific parameters, notably structured data in electronic health records (e.g., diagnostic testing results, vitals measurements, demographic information). The input data of this medical device software is not transformed by other tools or products. This medical device software contains one input source (i.e., data in electronic health records) but includes multiple interdependent data elements (e.g., demographic data, laboratory and diagnostic testing results, treatments). These structured, regular data inputs from known sources of expected uniform quality do not appear to introduce novel or altered risks as a result of introducing the software solution. An HCP would review the same data to make an independent decision if the software was not available.

Degree of Autonomy	<ul style="list-style-type: none"> •Is a user in the loop? Is the user in the loop a health care professional? 	<p>In terms of the Degree of Autonomy, a clinician is in the loop to review any outputs flagged by the software and to make the next decision in the clinical workflow – such as follow up tests for the patient. However, a clinician will not be informed of patients who have <i>not</i> met the threshold to be considered “at risk” by the software.</p>
Explainability of Software and Output (underlying logic including the algorithm/technology used, relevant development data characteristics, and how output is reached)	<ul style="list-style-type: none"> •Is the relevant functionality of the product explained and understood by the user? •Is the relevant functionality of the product explained and understood by users other than the patient? Is different information provided to different user groups or patients? •Is the relevant functionality partially explained or partially able to be evaluated by the user (e.g., output provided with saliency maps)? 	<p>Considering the Explainability of Software and Output, the relevant functionality of this medical device software is explained (i.e., within its indication for use and ordering requirements) and is understood/can be evaluated by the user (i.e., input data includes structured data elements in electronic health records). The analysis (i.e., statistical or computational approach) is partially explained to the user.</p>
Destination/Target of Output	<ul style="list-style-type: none"> •Is the output the only instruction/data/information needed to drive the target’s next action? 	<p>Considering the Destination/Target of Output, this software likely does not provide an output that would be the sole instruction/data/information to drive the HCP user’s next step. The output will present cases for the HCP to review and introduce a single new datapoint (patient has been identified as above a threshold). The HCP will have the patient’s data for review in addition to the information that the patient has exceeded the threshold to help inform their next decision. However, as noted above, the HCP will not be presented with any data on patients who do not exceed the software’s threshold, which could result in <i>no</i> decision made for such patients.</p>

After consideration of questions related to Device Function/Use, it may be considered that the output type, supplementing additional information to contribute to a clinical interpretation or workflow recommendation, in this case a prediction or diagnosis commonly accepted in clinical practice or based upon sound scientific principles, may not greatly impact the risk of the device. However, the specific threshold calculation is proprietary and the software is replacing a manual review of a patient record and introduces the possibility of incorrectly filtering patients for review by the HCP.

A.2.4 Device Change Management

Characterization Feature	Considerations for Medical Device Software Risk Characterization	Discussion
--------------------------	--	------------

Degree of Learning/Change Management Autonomy	<ul style="list-style-type: none"> •Does the medical device software independently change its underlying algorithms? •How often is medical device software performance verified? •Are updates to algorithmic performance driven by non-clinical or clinical users, or manufacturer driven, or a combination of these users? 	In considering the Degree of learning/change management autonomy, this medical device software does not independently change its underlying algorithms. The performance of this medical device software is verified on an annual schedule by the product developers and validated by clinical users within the specific healthcare site. Updates to the algorithmic performance are monitored by clinical users and the manufacturer.
Domain of Learning/Change Implementation	<ul style="list-style-type: none"> •Is domain specific implementation necessary to achieve adequate software performance? •Where are changes intended to be implemented and how variable are the domains? 	In Domain of Learning/Change Implementation, note that learning and/or change management may result in different accuracy or precision when this software is used across different clinical sites or regional locations (i.e., based on the demographic characteristics of the individuals in whom this software is used).
Installation, Update and Error Correction Infrastructure	<ul style="list-style-type: none"> •What specific channels are used to distribute the medical device software? •Does the medical device software have multiple installation locations? Where are corrections initiated? 	Regarding Installation, Update and Error Correction Infrastructure, note that this medical device software’s distribution channel is a web application, and that the software installation occurs on a server at the individual clinical site by clinical users.

In summary, for such a product, overall impact on risk posed or introduced- by the software takes into consideration multiple characterization features across information groupings, and those that are most relevant to the particular device software may be different depending on the device’s intended use/intended purpose. For this reason, it is critical to have a clear description of the software to help build an understanding of the role of the medical device software and its unique implementation. For this example device, the particular software solution may introduce risks related to the automation of a previously manual step and new failure points in the intended workflow. However, because of the device’s medical purpose and context of use, these risks may not have notably high impact. These considerations can be taken together when considering how the decision to design this software solution may impact the overall risk of the device or raise different hazards.

Appendix E: Examples Comparing Specific Risk Considerations

As with Example A above, addressing each of the characterization features through corresponding questions is helpful for evaluating risk. The below questions are listed by information grouping to support comprehensive discussion of risk considerations. As discussed above, while the questions below relate to evaluating the risk of the medical device, it is important that any regulatory assessment also considers the benefit to health from the use of the medical device and weighs the benefits against any risks of the medical device.

The pairs of comparative examples below further illustrate that the hazards to be extracted in risk analysis can differ based on the unique characteristics of a given medical device software.

Example 1: Software intended to provide a therapeutic experience to reduce and relieve pain.

Scenario 1.1: The software is intended to be used in conjunction with prescribed pain management medications to reduce and relieve pain in cancer patients undergoing chemotherapy.

Scenario 1.2: The software is intended to be used to reduce and relieve pain in osteoarthritis patients who cannot take other pain relief medication.

In both scenarios in example 1 above, the intended use of the medical device software is to provide therapy to reduce and relieve pain, where the cause of such pain (i.e., the Intended Disease or Condition) is not the primary distinguishing feature that contributes to understanding the risk of the medical device software. Rather, in this case, understanding whether the medical device software is intended to be used adjunctively (i.e., the Medical Purpose) contributes significantly to potential hazards considered in the risk analysis of the software.

In scenario 1.2, the software is meant to provide therapy for patients who cannot utilize other pain relief therapy. Because the software is itself intended as therapy and cannot be used with, or adjunct to, additional treatment, the risk of the software could be considered higher in scenario 1.2 than 1.1. The failure of the software output to provide efficacious therapy may be considered a single-point failure for achieving the intent of patient pain reduction or relief; and therefore, the intended medical purpose may contribute to the hazards considered in risk analysis more than the software used in conjunction with other therapy, described in scenario 1.1.

In these two scenarios for a similar medical device software, within the Medical Problem and/or Objective information grouping, characterization features contribute to the risk of the software differently. For such a product, the Intended Disease or Condition does not solely impact the risk posed by the software, but a more detailed understanding of the Medical Purpose contributes to a more complete understanding of the medical device software's risk.

Example 2: Software that aggregates data and highlights trends from a wearable monitor for patients diagnosed with heart failure

Scenario 2.1: The software is intended to aggregate data and highlight trends from a wearable monitor for patients diagnosed with heart failure to help patients monitor their risk of hospitalization. The software helps to provide simple data visualizations to better understand the patient’s longitudinal data, such as tracking an individual’s health, care usage, and outcomes over time.

Scenario 2.2: The software is intended to aggregate data and highlight trends from a wearable monitor for patients diagnosed with heart failure to help patients and their healthcare provider with longitudinal data about the patient’s heart health. The software provides simple data visualizations, including highlighting trends, to help the healthcare provider monitor their patient’s risk of hospitalization between regularly scheduled visits and could be used to inform treatment-related decisions.

In example 2 above, the intended user for the medical device software in scenario 2.1 is limited to patients seeking to obtain more information about their own condition. In scenario 2.2, healthcare providers are included in the intended user group and have access to the data in addition to the patient themselves. In this case, a health care professional has specialized training that provides them with additional context to understand the data and trends the medical device software is highlighting, which a patient may not have. For this reason, it might be considered that the Intended User in scenario 2.2 may reduce the likelihood of some of the related hazards considered in risk analysis compared to scenario 2.1, because at least one intended user in scenario 2.2 has expertise and training to appropriately understand and respond to the data they are receiving. The health care professional is provided access to the data such that it is not essential for the patient to independently identify if and when their data should be conveyed to their doctor.

However, it may also be worth considering that there is greater variability in the Intended Users of the medical device software in scenario 2.2 than scenario 2.1, because of the introduction of the clinician user. This difference also impacts the understanding of risk posed by the software, where the information must be conveyed adequately and appropriately to the different user groups. It is important to consider that multiple factors may influence the risk associated with any given characterization feature – a clinician or trained user does not always independently indicate a decrease or increase in applicable hazards in the risk analysis of a device.

Example 3: Software function that uses physiological data captured on a wearable consumer product to determine the severity of symptoms in a patient with Parkinson’s disease.

Scenario 3.1: The software is intended to aggregate measurements obtained from a regulated medical device and analyze to monitor the severity of symptoms such as tremor in a patient with Parkinson’s disease.

Scenario 3.2: The software is intended to aggregate measurements obtained from a wearable consumer product and analyze to monitor the severity of symptoms such as tremor in a patient with Parkinson’s disease.

In example 3 above, the Input Source in scenario 3.1 is limited to measurements obtained by a regulated medical device. In scenario 3.2, measurements are obtained by a wearable consumer product that is not subject to regulatory oversight as a medical device. In this case, the wearable consumer product may allow for expanded opportunities for collecting patient data. However, the aspects of the performance of the wearable consumer product may be outside of the control of the developer. For this reason, it may be considered that the Input Source in scenario 3.2 may pose more applicable hazards for risk analysis than in scenario 3.1, because the manufacturer developing the software may not have life cycle control over the source of the data it is analyzing to monitor the severity of symptoms. In this case, additional steps may be necessary for the manufacturer to monitor performance of the wearable consumer product and to communicate any changes in performance to the user. In contrast, scenario 3.1, which obtains measurements from a regulated medical device, benefits from the verification and validation needed to obtain authorization (in cases where the intended use is fit for purpose), which may reduce applicable hazards due to a greater accuracy and precision of measurements of a product developed for the intended use. Regulations applicable to software using consumer products to perform regulated device functions vary by jurisdiction.

**Please visit our website
for more details.**

www.imdrf.org

Disclaimer

© Copyright 2024 by the International Medical Device Regulators Forum.

This work is copyright. Subject to these Terms and Conditions, you may download, display, print, translate, modify and reproduce the whole or part of this work for your own personal use, for research, for educational purposes or, if you are part of an organisation, for internal use within your organisation, but only if you or your organisation do not use the reproduction for any commercial purpose and retain all disclaimer notices as part of that reproduction. If you use any part of this work, you must include the following acknowledgement (delete inapplicable):

"[Translated or adapted] from [insert name of publication], [year of publication], International Medical Device Regulators Forum, used with the permission of the International Medical Device Regulators Forum. The International Medical Device Regulators Forum is not responsible for the content or accuracy of this [adaption/translation]."

All other rights are reserved, and you are not allowed to reproduce the whole or any part of this work in any way (electronic or otherwise) without first being given specific written permission from IMDRF to do so. Requests and inquiries concerning reproduction and rights are to be sent to the IMDRF Secretariat.

Incorporation of this document, in part or in whole, into another document, or its translation into languages other than English, does not convey or represent an endorsement of any kind by the IMDRF.