Deadline of Submission of Comments: 27-Jan-20
Document Number: ASB BPR 114
Document Title: Best Practice Recommendation for Validation of Forensic DNA Software

Note: a specific Proposed Resolution must accompany each comment or it cannot be considered.

#	Section	Type of Comment (E-Editorial, T-Technical)	Comments	Proposed Resolution	Final Resolution
4	Gerneal	E	The term software is not defined in the document. This document can be interpreted as applying to software involved in DNA testing and analysis exclusively, when in fact it applies to other software such as LIMS and other tracking systems. Is there going to be a standard that accompanies this best practice document on software validation? If not the language in this document should be made stronger.	Consider reorganizing and editing the title of the document to Best Practice Recommendations for Validating Software Programs used in Forensic DNA Laboratories.	Accept with modification: The title was edited to say: Best Practice Recommendations for Internal Validation of Software used in Forensic DNA Laboratories.
13	Title	E	Add an 's' to the word "recommendation'	Best Practice Recommendations for Validation of Forensic DNA Software	Accept with Modification: However the title was edited "Best Practice Recommendations for Internal Validation of Software used in Forensic DNA Laboratories".
5	Foreword	E/T	ASB validation documents have made the distinction between developmental and internal validation, however; this distinction is missing from this document. If this document is meant to cover software programs developed in house, there are specific tests that should be used in the development of a software program and different tests for implementation. This document needs greater specificity beyond the term validation.	If there is a standard that accompanies this document, the specifics for developmental and internal validation should be included in that document. However, this document needs to have sections added to address the best practices for developmental and internal validation.	Accept with modification: The document was edited to clarify that it is for "internal validation" only. Also Section 3.4 was added in Section 3.
6	1	E/T	Is this best practice document meant to apply to CODIS software as well?	A statement like the one on probabilistic genotyping should be added to the document if CODIS software isn't applicable to the document. This should be added in the event there's a new laboratory looking to come online with CODIS.	Reject: CODIS software is applicable for this document.
14	1	E	Comma needed between words 'part' and 'or'	Software used as a component, part, or accessory of instrumentation.	Accept
15	3	T	Additional terms and definitions required	Add 'internal validation' and 'developmental validation' and define.	Accept with Modification: Section 3.4 was added to define "internal validation". Developmental validation is beyond the scope of this document therefore a definition is not required.
47	3.2	T	Whether software is characterized as "complex" is highly consequential to the rigor required during validation per this document. Yet, the definition of "complex software" is very vague (e.g. what qualifies as "many lines of code"?), and doesn't provide enough guidance to someone (in a forensic biology lab, or otherwise) who is attempting to apply it.	Provide a more precise definition of this term	Accept with modification: This definition was updated using language from the software engineering community. Also, section 4.8.2.3 wasupdated to reflect this change.
7	3.2, 4.8.2.3	E		Clarify what is meant by many lines of code.	Accept with modification: This definition was updated using language from the software engineering community, and the "many lines of code" language was deleted from this definition.
48	3.3	T	Whether software is characterized as "critical" is highly consequential to the rigor required during validation per this document. Yet, the definition of "critical software" requires making a judgment call about whether software "substantially influences" "any ... item deemed integral". These terms are unacceptably vague, and fail to provide real guidance to someone attempting to apply it	Define "substantially influences"	Accept with modification: This definition was updated and the "substantially influences" was replaced by "directly affect". Also, sections 4.8.2.1 - 4.8.2.2 were updated to reflect this change.
8	3.3, 4.8.2.3	T/E	The use of substantially in these two recommendations is not clear. With such a wide variety of software meant to be covered by this document and this serving as a best practice recommendation, specific examples should be provided.	Consider adding examples directly to the text of the document or creating a second annex where examples are given.	Reject with modification: This definition was updated and the "substantially influences" was replaced by "directly affect". Also, sections 4.8.2.1 - 4.8.2.2 were updated to reflect this change.
49	3.6 and 3.7	T	These two definitions would benefit from examples to clarify the difference between an "invalid" input and an "incorrect or inverse" input	Provide examples of "invalid" and "incorrect or inverse" inputs	Accept with Modification: 3.7 now includes this example "(e.g., inputting a letter when a number is required, and observing an error message)".
50	3.12	T	The definition of "risk assessment" is inappropriately circular and vague, given that the term "risk" is never defined. The term is also used in a way that is inconsistent with other sources. For example, ISO defines "risk" in the medical device context as: "combination of the probability of occurence of harm and the severity of that harm," "risk assessment" as "overall process comprising a risk analysis and a risk evaluation," "risk analysis" as "systematic use of available information to identify hazards and estimate the risk," and "risk evaluation" as "process of comparing the estimated risk against given risk criteria to determine the accepability of the risk."	Provide a complete definition of all the terms encapsulated in the term "risk assessment," specifically including actually defining "risk"	Accept: General definition of "Risk" added

#	Section	Type of Comment (E-Editorial, T-Technical)	Comments	Proposed Resolution	Final Resolution
9	3.12	T/E	The definition is a circular definition.	An alternate word for risk (error, hazard) should be used in the sentence.	Accept: General definition of "Risk" added
16	3.14	E	Change 'ex.' to 'e.g.' and move comma after 'e.g.'	(e.g., add-ins/plug-ins and macros).	Accept (now 3.15)
10	3.15	E	The term is not used in the document outside of section 3.	Incorporate the term or remove it from the document. Finding places to add this in the document are difficult since there is a definition for reliability testing.	Accept: 3.15 was deleted from section 3.
11	3.17		The term is not used in the document outside of section 3.	Software can be added to 4.8.3.1 ahead of "testing types", after "functional" in 4.8.3.1a, and ahead of "tests" in 4.8.3.2.	Accept
17	3.18	T	Add 'internal' to term 'software validation'	internal software validation (to differentiate from developer validations)	Accept
1	4.1	E	Missing items from list under 3.3 (casefile documentation, accuracy of results, report wording). If intentional, clarification would be helpful to why these software are not included under this section.	Add the additional software if this was not intentional for consistency of the list that this documentation applies to.	Accept
51	4.1	T	It would be inappropriate to use software that impacts the integrity of evidence, etc, in casework before it has been validated. Thus, the word "should" should be upgraded to "must" to communicate that this is essential	Change "should" to "must"	Reject: ASB guidelines recommend the use of the word "should" for recommendations and guidelines. Must usually refers to a requirement. See updated forward section.
52	4.1	E	The word "where" doesn't make sense in the sentence beginning "There may be examples…"	Change "where" to "that" in the sentence beginning "There may be examples…"	Reject: "Where" is the preferred term.
53	4.1	T	In addition to documenting the extent to which they rely on modules they are unable to validate, the lab should verify and document developmental validation of those modules, and ensure that the developmental validation establishes the reliability of the module. This review must be conducted by/in consultation with someone with appropriate expertise (see comment for 4.3)	Add a sentence requiring the lab to verify and document developmental validation of any modules they are unable to validate, and ensure that the developmental validation establishes the reliability of the module. This review must be conducted by/in consultation with someone with appropriate expertise (e.g. computer science/software engineering)	Reject: The entire document was edited to clarify that it covers internal validalidation only.
18	4/4.1	E	Comma needed between words 'interpretations' and 'and/or'	the interpretations, and/or statistical conclusions	Accept
19	4/4.2	E	Add "e.g.," to list in parentheses	(e.g., operating system, database management system, etc.).	Accept with modification. "e.g.," added, but "etc." removed as both are not necessary.
20	4/4.2	E	Comma needed between words 'example' and 'substantive'	(for example, substantive…	Accept
21	4/4.2	E	Suggest grammar and punctuation change	the consequences must be evaluated and a software risk assessment and validation should be conducted if applicable.	Accept with modification. Paragraph now reads: As the computing environment of the software evolves (for example, substantive version changes to the operating system, or fundamental changes to the computing hardware architecture), the consequences shall be evaluated and a software risk assessment shall be performed. An internal validation should be conducted if applicable based on the outcome of the software risk assessment.
54	4.2	T	This section requires evaluation of consequences of computing environment evolution "and if applicable, a software risk assessment and validation should be conducted." It needs to explain/clarify when risk assessment and validation "is applicable"	Add a sentence explaining/clarifying when risk assessment and validation "is applicable"	Accept with modification: The second paragraph in this section was edited for clarity. Paragraph now reads: As the computing environment of the software evolves (for example, substantive version changes to the operating system, or fundamental changes to the computing hardware architecture), the consequences shall be evaluated and a software risk assessment shall be performed. An internal validation should be conducted if applicable based on the outcome of the software risk assessment.
	4.3	T	Whether the lab itself is conducting the software validation, or is bringing in a third party, there should be a requirement that the individual conducting the validation have the requisite expertise to validate software -- a different skillset than validating DNA methodologies	Add a sentence requiring the individual conducting the validation to have the requisite expertise to validate software (e.g. computer science or software engineering background)	Reject: The qualifications of the individual performing the validation are outside of the scope of this document.

#	Section	Type of Comment (E-Editorial, T-Technical)	Comments	Proposed Resolution	Final Resolution
22	4.3	T	Both approaches would probably incorporate documentation of the testing done by the developer.	4.3 The forensic laboratory should choose one of the two following internal validation approaches: 4.3.1 The forensic laboratory performs an internal validation of the software. 4.3.2 The forensic laboratory uses a third party to perform the internal validation. Both approaches may also include documented developmental validations performed by the developer, but not to the exclusion of an internal validation.	Reject: The entire document was edited to clarify that it covers internal validalidation only.
2	4.3.1	E	Remove period between words 'developer' and 'but'	Delete out period	Accept
56	4.3.1	E	The period after "developer" should be a comma	Change the period after "developer" to a comma	Accept with modification: Period removed. No need for a comma.
57	4.3.1	T	The lab must not utilize a developer's validation without scrutinizing it with a critical eye. Any validation by a developer should be evaluated by the lab to ensure that it appropriately establishes reliability of the software. This review must be conducted by/in consultation with someone with appropriate expertise (see comment for 4.3)	Add these sentences at the end of this section: "Further, any validation performed by a developer should be evaluated to ensure that it appropriately establishes reliability of the software. This review must be conducted by/in consultation with someone with appropriate expertise (e.g. computer science/software engineering)."	Reject: The qualifications of the individual performing the validation are outside of the scope of this document.
23	4.4	E	Add appropriate punctuation after the word 'Note'	NOTE:	Reject: The Note follows ASB's style of formatting.
24	4.4	E	Add appropriate punctuation after the word 'use' and 'case'	for every use, case, or scenario	Accept with Modification: "Use-case" is one word and a comma was added after case.
58	4.4	T	"operational environment" should be defined	Define "operational environment" here or under Terms and Definitions	Reject: Operational environment is synonymous with laboratory.
59	4.4	T	The note states that software associated with an instrument may be validated in conjunction with the instrument. It should also make clear that this does not change the requirements of software validation or the rigor with which it must be conducted	Add a sentence at the end of the note: "However, this does not change the requirements of software validation or the rigor with which it must be conducted."	Reject: There is no implication that this should be less rigor in the note as stated.
60	4.5	T	The purpose of this section is unclear. Explain functionality to whom, and for what purpose? Is this section essentially communicating that the lab should go to the developer and ask them to explain how the software functions to facilitate the lab conducting its validation? If so, this suggests a third party should be brought in to conduct the validation. It's inappropriate to have persons without expertise rely heavily on the developer in crafting its own validation without independent understanding of the underlying system	Clarify the purpose/meaning of this section. Further, reiterate the importance of having someone with appropriate expertise conduct these validations	Reject: 4.5.1. and 4.5.2 explan the purpose of this section.
25	4.5.1	E	Add appropriate punctuation after the word 'documentation' and 'manual'	documentation, such as a user's manual,	Accept
26	4.5.2	T	Grammar and punctuation	The laboratory may rely on the results of testing conducted by the developer, but the laboratory should extend testing during the internal validation.	Accept with Modification: "internal" was added to the sentence. "Tests" is appropriate as is.
61	4.5.2	T	In 4.3.1, this guidance appropriately points out that developer's validations "in themselves, are not sufficient". Thus, this section should require (i.e. "must", not "should") the laboratory (or third party) to extend tests during validation. Also, there should be a requirement that any developmental validation relied upon by the lab be thoroughly evaluated by someone with appropriate expertise	Change "should" to "must". Add sentence: "Further, any developmental validation relied on by the laboratory should be evaluated by/in consultation with someone with appropriate expertise (e.g. computer science, software engineering), independent of the developer, to ensure that it appropriately establishes reliability of the software."	Reject: Per ASB Style Manual "should" is the preferred word for guidelines. See updated Foreword section for further information. The purpose of this documentation is to describe the validation and not the qualifications of the individual(s) performing the validation.
62	4.6.2	T	Heavy reliance on the developer to shore up holes in the lab's knowledge base -- in lieu of consulting with someone with appropriate expertise to conduct software validations -- is a real concern. There appears to be a pattern of heavy reliance on the developer throughout these recommendations (e.g. 4.5). Again, there should be a requirement that someone with appropriate expertise -- and independent of the developer -- be involved in the lab's validation	Add sentence: Any tests recommended by the developer should be evaluated by someone independent of the developer, who has appropriate expertise (e.g. computer science, software engineering)	Reject: The purpose of this documentation is to describe the validation and not the qualifications of the individual(s) performing the validation.
63	4.7	T	It is unclear whether the developer-designed studies mentioned here refer to studies conducted by the developer or the recommended tests under 4.6.2. To the extent it is the former, the "and/or" puts this section in conflict with sections 4.3.1 and 4.5.2.	Clarify the meaning as set forth in the comment, and if this section is intended to include studies conducted by the developer, change "and/or" to "and" to bring into conformance with sections 4.3.1 and 4.5.2	Accept with modification: This section was clarfied by removing "and".

#	Section	Type of Comment (E-Editorial, T-Technical)	Comments	Proposed Resolution	Final Resolution
27	4.7 including 4.7.1, 4.7.2, & 4.7.3	T	Are these requirements or recommendations? Statements that include the words 'should', 'may' or 'can' are recommendations rather than requirements.	(4.7.1, 4.7.2) the forensic laboratory is responsible for determining that the validation design, testing procedures, and documentation meet the recommendations outlined in this document. (4.7.3) If these recommendations have not been met	Accept: "Requirements" was replaced with "recommendations" in sections 4.7.1, 4.7.2 and 4.7.3.
29	4.8.2	T	Add clarity to tie in subsections 4.8.2.1, 4.8.2.2, 4.8.2.3, 4.8.2.4 and combine subsection 4.8.2.5 with 4.8.2	A risk assessment should be performed to make an objective assessment of the level of criticality (i.e. critical or not critical) and complexity (i.e. complex or not complex) of a software program or module. A risk assessment should be conducted whenever a new version of a software program or module is released or when there are software modifications or updates.	Accept with modification: The definitions for " Complex", "Critical", "Risk assessment" were updated and the definition for "Risk" was added to this document. The risk assessment was edited for clarity and section 4.8.2.5 was deleted from the document.
32	4.8.3	T	How does Figure 2 tie into the document?	Move Figure 2 to a more appropriate place within the document and reference it within the dialogue of that section. Don't break up section 4.8.3 with Figure 2.	Accept: Table moved and referred to from section 4.8.3 to section 4.8.3.2.
64	4.8.2 and all subparts except 4.8.2.5 (i.e. 4.8.2.1-4.8.2.4)	T	A risk assessment appears to be a crucial part of software validation, because it results in software being classified as complex (or not), and crucial (or not). This, in turn, impacts the level of validation required by this document. Yet, there is no guidance on how to conduct a risk assessment. Further, the definitions of "critical", "not critical", "complex", and "not complex" are too vague to be useful or meaningful (see comments to 3.2 and 3.3). Whenever the lab (or third party) is in doubt, they should always err toward higher levels of validation, not lesser. The risk assessment should be carried out by someone with appropriate expertise.	Explain what is required of a risk assessment. Define terms in 4.8.2.1-4.8.2.4 with enough precision to be meaningful. Add a sentence communicating that when in doubt, err toward higher levels of validation. Include a requirement that the risk assessment be conducted by someone with appropriate expertise (e.g. computer science or software engineering background).	Accept with modification: The definitions for " Complex", "Critical", "Risk assessment" were updated and the definition for "Risk" was added to this document. The risk assessment was edited for clarity and section 4.8.2.5 was deleted from the document.
28	Figure 1	E	Move Figure 1	4.8 Validation should follow a predefined plan as depicted in Figure 1. [Insert Figure 1.] Don't break up the sub-sections of 4.8; it is confusing.	Accept: Figure 1 moved after 4.8.
30	4.8.2.2	E	Add appropriate punctuation after the word 'conclusions'	process, conclusions, or documentation.	Accept
31	4.8.2.4	E	Add appropriate punctuation after the word 'module'	program, module, or upgrade	Accept
33	4.8.3	T	The required test types; required or recommended? Statements that include the words 'should', 'may' or 'can' are recommendations rather than requirements.	The recommended test types...	Accept with modification: "recommended" replaced "required".
34	4.8.3.1	E	Add appropriate punctuation after the word 'validation'	...be performed during validation:	Reject: Punctuation is correct as is.
35	4.8.3.1	E	Remove punctuation after all instances of the word 'testing'	1) positive testing 2) negative testing 3) boundary testing 4) fuzz testing b) Reliability Testing c) Regression Testing	Reject: Punctuation is correct as is.
36	4.8.3.2	E	Add appropriate punctuation after the word 'levels'	following tests, at a minimum, are recommended for given criticality/complexity levels:	Reject: Punctuation is correct as is.
37	4.8.3.2	E	Remove comma after '(e.g.,'	(e.g. already validated software with a corrected misspelling).	Reject: Punctuation is correct as is.
38	4.8.3.2	E	Remove punctuation from multi-level list; Remove punctuation after instances of 'e.g.' or 'i.e.' when only a single example or "that is" follows.	a) Not Critical/Not Complex (e.g. already validated software with a misspelling corrected) 1) Functional testing: Positive b) Critical/Not Complex 1) Functional testing: Positive, Negative, Boundary 2) Regression testing: If applicable (i.e. software updates) 3) Reliability testing: If applicable (e.g. multi-user environment) c) Not Critical/Complex (e.g. updates to module with quality flags used to aid analysis) 1) Functional testing: Positive, Negative, Fuzz 2) Regression testing: If applicable (i.e. software updates) d) Critical/Complex 1) All Functional, reliability, and regression testing	Accept with modification: Punctuation is correct as is. Only the suggestion regardign section c) was accepted. Comma was added after reliability in d-1).
3	4.8.3.3.2	E	It is confusing that probabilistic genotyping systems are included here. The scope includes the statement: 'This document does not cover probabilistic genotyping'.	Remove the (e.g., probabilistic genotyping)	Accept: Section 4.8.3.3.2 was removed in its entirety.

#	Section	Type of Comment (E-Editorial, T-Technical)	Comments	Proposed Resolution	Final Resolution
12	4.8.3.3.2	E	Probabilistic genotyping should be removed as the example since the document does not cover probabilistic genotyping.	Provide another example. I'm not sure if the Cal DOJ Y-STR Mixture filter or CODIS software applies here.	Accept: Section 4.8.3.3.2 was removed in its entirety.
39	4.8.3.3.2	E	Remove comma after '(e.g.,'	(e.g. probabilistic genotyping systems)	Accept: Section 4.8.3.3.2 was removed in its entirety.
65	4.8.3.3.2	T	Why does this only apply to software that is deemed complex and critical? It seems to be in tension with 4.8.3.3.3, which requires different data sets when setting parameters/boundaries and testing the system (which relates to software categories other than complex + critical)	Bring sections 4.8.3.3.2 and 4.8.3.3.3 into concordance	Accept: Section 4.8.3.3.2 was removed in its entirety.
66	4.8.3.3.2	T	This document states that it does not apply to probabilistic genotyping software. Given that, it would be more helpful to people applying this standard if an example of complex, critical software was provided that actually falls within the purview of this document	For the "e.g.", use a software program that falls within the purview of this document	Accept: Section 4.8.3.3.2 was removed in its entirety.
40	4.8.3.3.4	E	Remove comma after the word 'released'	Whenever a new version of the software program or module is released or when there are software modifications or updates,	Reject: The sentence reads well as it is.
41	4.8.3.4	T	Sections 4.8.3.3.4, 4.8.3.4, & 4.8.3.4.1 seem redundant.	Remove redundant recommendations and/or combine with first mention.	Reject: These sections add value to this document.
67	4.8.3.4	T	This section should require the lab (and particularly someone with relevant expertise) to review the developer's validation with a critical eye; should require the lab to perform its own testing (or have a third party conduct testing at the lab)-- i.e. "should" become "must"; and should specify which tests the lab/third party must conduct	Revise to require the lab (and particularly someone with relevant expertise) to review the developer's validation with a critical eye; change "should" to "must" to reflect that testing in the lab is essential; and should specify which tests the lab/third party must conduct	Reject: This document is a Best Practice Recommendation (not a standard with requirements) and the term" should" is best suited for use. Please refer to the updated Forward section as well.
68	4.8.3.4.1	T	"The forensic lab should additionally conduct some of their own tests" is extremely vague. What tests? As with other sections above (e.g. 4.6.2), this section reflects an overreliance on the developer	Revise to set forth specific tests that the lab/third party should conduct. Add sentence: Any test sets recommended by the developer, or built into the software should be evaluated by someone independent of the developer, who has appropriate expertise (e.g. computer science, software engineering).	Reject: The types of tests vary greatly from each software package. Therefore, this recommandations are generic in nature.
42	4.8.5.1	E	Suggest grammar and punctuation change	Failing some of the tests does not necessarily mean invalidation. The laboratory may decide that some failures represent minor inconveniences that do not invalidate the software. Alternatively, the laboratory may decide that a single critical or a combination of moderate failures is intolerable and the software cannot be accepted.	Accept
69	4.8.5.1	T	This section is extremely problematic. It appears to give complete discretion to the lab to decide whether a failure of the software renders it invalid. It provides no guidance for making this determination. It appears to be unnecessary and antithetical to a best practice document.	Delete this section	Reject: Section 4.8.5.1 in combination with the subsections describes the process in full for the documentation and handling of failures.
43	4.8.6.1.1	E	Remove punctuation from multi-level list	a) organization that conducted testing b) defined user requirements c) risk assessment decision and reasoning d) software name including version number e) test cases: 1) date and time of test(s) 2) operating system 3) input data f) acceptance criteria g) test result(s) h) record of any system failure and actions taken to ensure failure was resolved i) record of formal acceptance or rejection j) date the software was approved for use, if implemented	Reject: Punctuation is correct as is.
44	From CB Ballot		The concepts in this document appear to be applicable to probabilistic genotyping software and thus I would recommend removing the sentence under the scope that states: "This document does not cover probabilistic genotyping." I recognize that other documents have and will likely go into further detail on PGS validation but I think it will create confusion to have a document on best practices forensic DNA software validation that does not include PGS.		Accept with modification: See first paraghraph in section 5

#	Section	Type of Comment (E-Editorial, T-Technical)	Comments	Proposed Resolution	Final Resolution
45	From CB Ballot		I'm not really sure of the purpose of this document. If software validation is important enough to warrant a BPR, I would think a standard with more directives needs to exist as well. Software that is critical to any assay or testing method should be covered in the validation of the method or assay and should not overlap with this document. The amount of software this document is supposed to cover is too much.		Reject: This is a best practice recommendation and what is being recommended is within the scope of this document.
46	From CB Ballot		Figure 2 appears redundant and too simplistic for the document		Reject: The figure is supposed to be used in conjuction with the text in section 4.8.3.2.