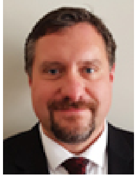
A step-by-step process to 95% autoverification

Bill Marquardt, C(ASCP), MBA, CSSBB

December 2015—Many laboratories have yet to reap the benefits of autoverification even though there is clear evidence of its benefits. During a recent internal study at Labsco, we discovered that more than 70 percent of our customers have not yet implemented AV. Of the customers who did perform AV, the highest percentage seen was about 60 percent (primarily in hematology and coagulation).



Marquardt

Three primary reasons were cited for the lack of AV implementation: no or limited technical resources to dedicate to the project; information system incapable of performing complex rules and results review; and lack of AV experience and an established AV implementation process. For more than 10 years, I have built AV algorithms and written rules for hundreds of analyzers. During that time, I have designed and honed a process that can be used to implement autoverification in a laboratory of any size. In this article, I will outline that process.

Any level of AV will result in quality and efficiency improvements. Many laboratorians believe that implementing an AV process is a one-time operation when, in fact, the laboratory should continue to refine the algorithms and rules to achieve as much AV as possible. Initial levels of AV in the 60 percent range are relatively easy to achieve (with the right process) and will result in significant improvement, but no lab should settle at this level. With the right set of algorithms and the right tools, it is possible to achieve levels greater than 95 percent.

I have found that although departments can vary widely in specifications, analyzers generally (with some exceptions) have the same basic data outputs, making it possible to some degree to standardize the algorithms required to build AV. Many laboratories begin by selecting the tool first, or simply assume their current laboratory information system or middleware product is up to the job. But not all LIS or middleware systems are created equal. All successful projects begin with specifications, and the right tool(s) can be selected only when the specifications are complete.

SMART TESTS is an autoverification design technique that can be used as a guideline to build an AV system. SMART TESTS is an acronym that stands for Scope, Master algorithm template, Analyzer-specific algorithm template, Reagent-specific algorithm, Tool selection and implementation, Translation of algorithms into rules, Exceptions review process, Substantiation (validation), Train, and Scale.

Scope. Step No. 1 is to define the project team. The point person and project champion for the AV team is generally the lab manager or lab director who is responsible for ensuring the project stays on task. Another critical component is a project manager. Most laboratories do not have access to a dedicated project manager so it is usually necessary to assign someone dual roles. The project manager should, if possible, be someone with project management background.

The primary subject matter expert should be the clinical pathologist (or PhD-level director if a specialized title exists for the department in question). Because of the impact AV will have on the laboratory, and the need to optimize and validate continually, it is best to assign an “autoverification expert” for the laboratory. This person can initially double as the laboratory/IT liaison, but as AV increases and the need for skilled labor on the bench decreases, a dedicated resource can be allocated.

The technical “team lead” and at least one department clinical laboratory scientist should also be on the team. These personnel will sometimes be different for different departments depending on the laboratory’s size. Finally, there should be one representative from information systems.

Prior to initiating the autoverification, weekly meetings may be appropriate. During AV implementation, members should meet daily (or at least three times per week). After implementation, it is critical to meet at least once per month to discuss enhancements, review metrics, and plan the next continuous improvement event.

Measuring key metrics before and after implementation is critical to understanding return on investment and monitoring for continuous improvement. Turnaround time, technologist utilization, and error rates are three metrics that should be understood and tracked. It is relatively uncommon for many laboratories to have access to all of these statistics but important for them to be obtained. TAT is a relatively easy metric to obtain from laboratory information systems, but technologist utilization (amount of time a technologist spends performing duties) is generally unknown. Error rates can be obtained by determining the number of corrected reports (due to releasing or data-entry error) per number of patients that are run on the analyzers where AV will be performed.

Complexity of AV algorithmic design and analyzer capabilities must be analyzed and weighed accordingly. Many laboratories begin AV in hematology because there are many common and somewhat standardized rules for review of hematology results. Algorithms for the AV of chemistry results tend to be more complex than most others. Balancing complexity of implementation with the potential gains will help in choosing the first AV project. Think about your project in terms of return on total investment.

Master algorithm template. All reagent-specific (or test-specific) AV algorithms must be derived from an analyzer-specific algorithm, which must be derived from a master algorithm. The master algorithm provides the high-level description that will act as the blueprint for the entire AV system.

It is best to use standardized flow chart symbology for all algorithms. The guidelines for basic flow chart diagrams can be found online, but the three most important symbols are a solid box (indicating a process), a double oval (indicating a start and termination), and a diamond (indicating a decision point or question). All decision points must be a yes or no question; no open-ended questions are acceptable. Each process, decision point, or termination should be numbered. Automated and manual processes should be outlined on the algorithm, perhaps with separate colors or symbols.

Maintaining version control for all algorithms is critical. Quick modification to a reagent-specific algorithm may be required and thus a change to the analyzer and master algorithms would also have to occur. Maintaining reference to the appropriate version of analyzer and master algorithms ensures that the laboratory does not necessarily need to change all operationally effective algorithms and rules currently in place, and it ensures compliance during inspection. Rules found in the “tool” (described later) that do not reference an algorithm version with the appropriate step should be suspect.

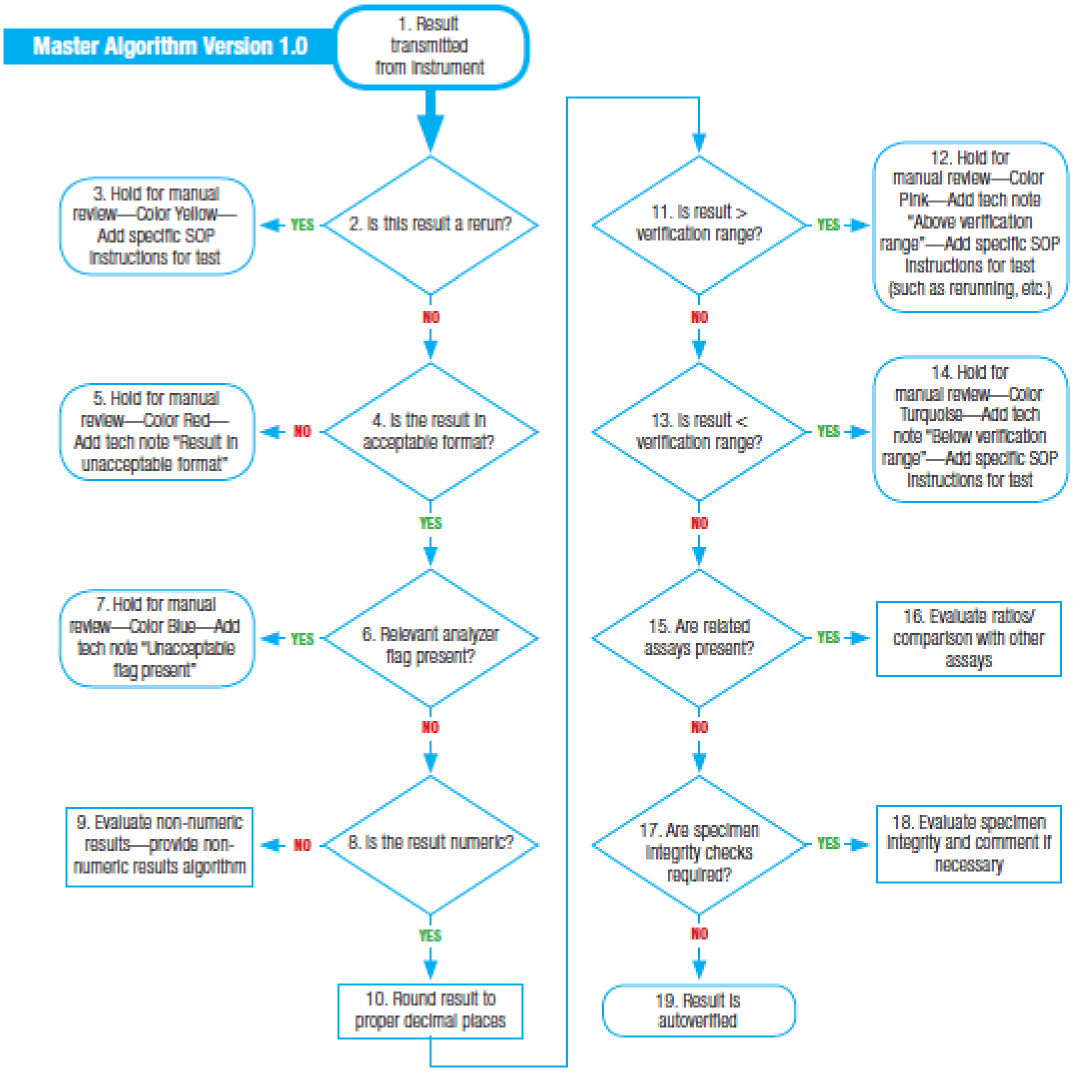

Master Algorithm Version 1.0
1. Result transmitted from instrument
2. Is this result a rerun?
YES
3. Hold for manual review—Color Yellow—Add specific SOP instructions for test
NO
4. Is the result in acceptable format?
NO
5. Hold for manual review—Color Red—Add tech note "Result in unacceptable format"
YES
6. Relevant analyzer flag present?
YES
7. Hold for manual review—Color Blue—Add tech note "Unacceptable flag present"
NO
8. Is the result numeric?
NO
9. Evaluate non-numeric results—provide non-numeric results algorithm
YES
10. Round result to proper decimal places
11. Is result > verification range?
YES
12. Hold for manual review—Color Pink—Add tech note "Above verification range"—Add specific SOP instructions for test (such as rerunning, etc.)
NO
13. Is result < verification range?
YES
14. Hold for manual review—Color Turquoise—Add tech note "Below verification range"—Add specific SOP instructions for test
NO
15. Are related assays present?
YES
16. Evaluate ratios/ comparison with other assays
NO
17. Are specimen integrity checks required?
YES
18. Evaluate specimen integrity and comment if necessary
NO
19. Result is autoverified

Fig. 1. Example of a simple master algorithm

The master algorithm should always start with results transmitted from an analyzer (**Fig. 1**). The first decision point is to determine if the result is a rerun (the same test was run on that patient at a previous point, typically on the same specimen). The reason a rerun check is performed first is that several processes later in the algorithm will result in a rerun being performed, and if a rerun check is not performed first, an endless loop situation could occur. Furthermore, decision points later in the algorithm will most likely be different for a rerun versus an initial result. Many analyzers will indicate if the result is a rerun (via a flag), but if a user manually orders the rerun on the analyzer, a process should be put in place to ensure reruns are identifiable by the rules. The laboratory should develop a rerun algorithm separately. For initial AV algorithms, it is acceptable to put a manual process in place to verify that the initial result and second result are within a certain percentage or nominal unit from the original. The algorithm could refer to an SOP already in place but ideally could provide direction to the technologist on a particular manual action to perform. Again, AV algorithms will be honed over time and an automated algorithm can be added later. Reruns are usually a small percent of the entire volume.

The second decision point in a master algorithm is to determine if the result is in the proper format. Certain results are expected to be numeric, others to be alphanumeric. Most analyzers send results greater than and less than the linearity range in an alphanumeric format. The analyzer vendor can provide the exact nomenclature for the expected formats. It is important to know exactly how the result will be transmitted (with the proper spaces and decimal places). This is a rule-out decision point, meaning if the result doesn't fall into one of the expected formats, it is an exception. Again, a manual process to manage these exceptions is a good place to start.

The third decision point in the master algorithm is a check for relevant analyzer flags. Analyzer flags are considered relevant if the result could be considered suspect. For example, an “H” flag could only mean that the result is above the analyzer's programmed reference range and therefore wouldn't be a relevant analyzer flag. A “LIN” flag could indicate a linearity issue with the reaction and thus a suspect result. The outcome of a relevant analyzer flag would be an algorithm outlining the actions required for certain analyzer flags. Again, a decent starting point would be to add this to the exception list to be reviewed manually. But addressing and adding actions to analyzer flags is somewhat straightforward since most analyzers have documentation detailing an action to be performed based on the flags.

The next few decision points focus on numeric versus non-numeric results. First we must determine if the result is a number. If the result is numeric, it should first be rounded to the correct number of places (unless the algorithm is such that rounding should be performed at a later stage). Next, we can compare it against a “verification range.” For initial AV algorithms, stopping the algorithm to review and take manual action is sufficient. But automating actions, such as sending a message to an analyzer to autodilute a sample due to a very high result, should be implemented where possible. If the result is not numeric, we must outline the actions to be performed on the expected non-numeric ranges. It is at this point where we could also add a delta check comparison. A delta check is a comparison of this result with a previous result on the same patient from a different draw (or visit). Although delta checking has somewhat fallen out of favor with many laboratorians, it should still be considered. Delta checking is useful to determine if a result is significantly different from a previous draw and thus would be suspect. It could also be used to eliminate the need for reruns if a patient is known to run abnormal. Delta checking must take into consideration a relevant time period.

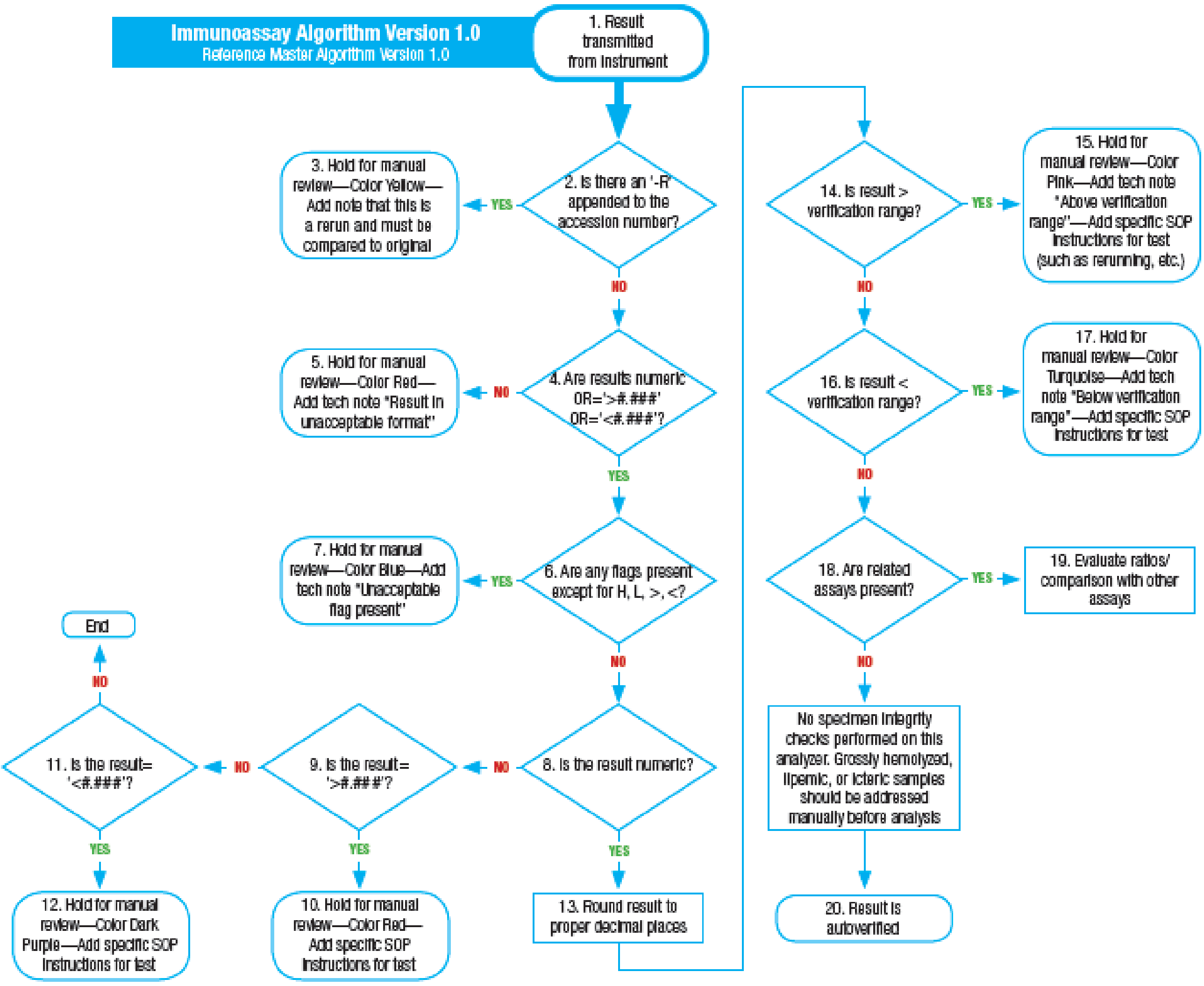

Immunoassay Algorithm Version 1.0
Reference Master Algorithm Version 1.0
1. Result transmitted from instrument
2. Is there an '-R' appended to the accession number?
YES
3. Hold for manual review—Color Yellow—Add note that this is a rerun and must be compared to original
NO
4. Are results numeric OR='>#.###' OR='<#.###'?
NO
5. Hold for manual review—Color Red—Add tech note "Result in unacceptable format"
YES
6. Are any flags present except for H, L, >, <?
YES
7. Hold for manual review—Color Blue—Add tech note "Unacceptable flag present"
NO
8. Is the result numeric?
NO
9. Is the result= '>#.###'?
NO
11. Is the result= '<#.###'?
NO
End
YES
12. Hold for manual review—Color Dark Purple—Add specific SOP instructions for test
YES
10. Hold for manual review—Color Red—Add specific SOP instructions for test
YES
13. Round result to proper decimal places
14. Is result > verification range?
YES
15. Hold for manual review—Color Pink—Add tech note "Above verification range"—Add specific SOP instructions for test (such as rerunning, etc.)
NO
16. Is result < verification range?
YES
17. Hold for manual review—Color Turquoise—Add tech note "Below verification range"—Add specific SOP instructions for test
NO
18. Are related assays present?
YES
19. Evaluate ratios/ comparison with other assays
NO
No specimen integrity checks performed on this analyzer. Grossly hemolyzed, lipemic, or icteric samples should be addressed manually before analysis
20. Result is autoverified

Fig. 2. Example of an analyzer-specific algorithm

The last few decision points in the master algorithm should focus on related tests and specimen integrity. For instance, direct bilirubin should never be higher than total bilirubin, and albumin should never be higher than total protein. In these cases, actions should be considered on all assays affected (for example, hold both the albumin and total protein for manual verification). Many analyzers also are capable of testing for specimen integrity factors such as hemolysis, icterus, and lipemia. It is at this point that the levels should be evaluated and a comment added to the test or, in extreme cases, the test result invalidated. When instruments cannot analyze for specimen integrity, and a test may be affected by higher levels of interferents, a manual process prior to analysis should be implemented as a check.

The master algorithm should contain all potential decision points that may be included in any reagent-specific algorithm. In the case where a specific reagent algorithm does not require a specific decision point, it can be excluded but with a specific notation. If a result passes all decision points for suspect results, it is autoverified. Again, initial AV algorithms do not need to be perfect, but they do need to ensure that any potential suspect results are caught as exceptions.

These algorithms do not include quality control decision points. Most laboratories continue to run QC prior to patient samples and this method ensures that all QC is passed prior to the AV algorithm. In my experience, the operational inefficiencies associated with failed QC when running in a random-access mode overrule its advantages. QC should continue to be run before patient sampling. As average of patient means (or patient moving averages) becomes more prevalent, AV should be suspended if and when thresholds are met. This discussion is outside the scope of this article.

Analyzer-specific algorithm template. After the master algorithm template has been created, analyzer-specific algorithms then can be derived for each model of analyzer. Analyzer behavior drives the specifics of the decision points from the master, such as analyzer-specific rerun behavior, error flags, and ability (or inability) to produce specimen integrity checking like lipemia, hemolysis, or icterus measurements. Department-specific SOPs that have been defined around an analyzer's functionality will define how the decision points within the algorithm are executed. Most likely, bench-level processes will be modified to best use AV due to analyzer-specific functionality, AV tools, and overall need for optimization. For example, a particular analyzer may or may not have onboard rerun capability. Without onboard rerun it might be necessary to manually order a rerun on the analyzer and append an "-R" after the accession number (or, for example, a "-10×" for a 10× dilution). The creative methods developed and applied would have to be a function of the ability of the AV tool to execute on the decision points.

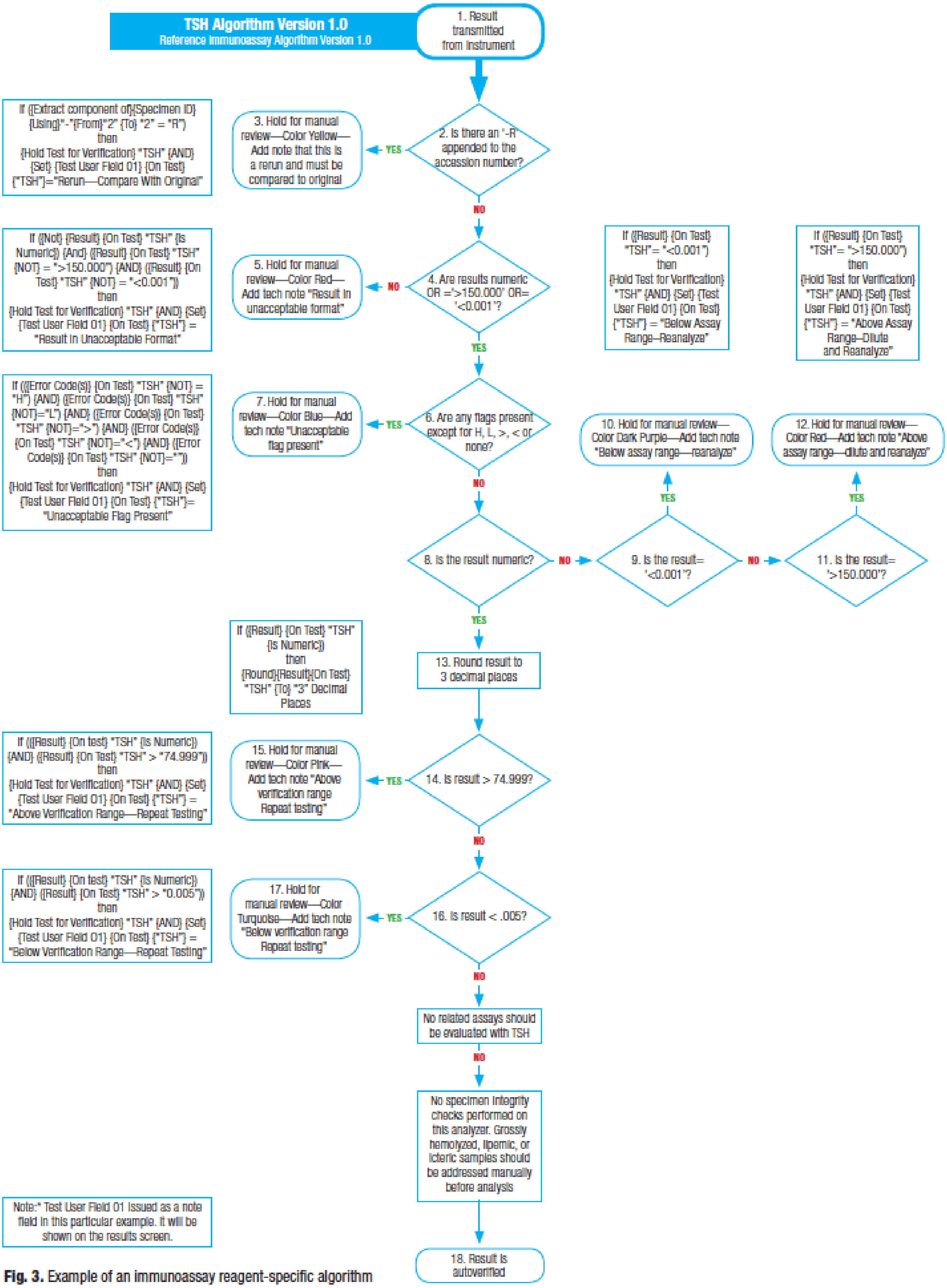

TSH Algorithm Version 1.0
Reference Immunoassay Algorithm Version 1.0
1. Result transmitted from Instrument
2. Is there an '-R' appended to the accession number?
YES
3. Hold for manual review—Color Yellow—Add note that this is a rerun and must be compared to original
If ({Extract component of}{Specimen ID} {Using}"-"{From}"2" {To} "2" = "R") then {Hold Test for Verification} "TSH" {AND} {Set} {Test User Field 01} {On Test} {"TSH"}="Rerun—Compare With Original"
NO
4. Are results numeric OR ='>150.000' OR= '<0.001'?
NO
5. Hold for manual review—Color Red—Add tech note "Result in unacceptable format"
If ({Not} {Result} {On Test} "TSH" {Is Numeric}) {And} ({Result} {On Test} "TSH" {NOT} = ">150.000") {AND} ({Result} {On Test} "TSH" {NOT} = "<0.001") then {Hold Test for Verification} "TSH" {AND} {Set} {Test User Field 01} {On Test} {"TSH"} = "Result In Unacceptable Format"
YES
6. Are any flags present except for H, L, >, < or none?
YES
7. Hold for manual review—Color Blue—Add tech note "Unacceptable flag present"
If (({Error Code(s)} {On Test} "TSH" {NOT} = "H") {AND} ({Error Code(s)} {On Test} "TSH" {NOT}="L") {AND} ({Error Code(s)} {On Test} "TSH" {NOT}=">") {AND} ({Error Code(s)} {On Test} "TSH" {NOT}="<") {AND} ({Error Code(s)} {On Test} "TSH" {NOT}="")) then {Hold Test for Verification} "TSH" {AND} {Set} {Test User Field 01} {On Test} {"TSH"}= "Unacceptable Flag Present"
NO
8. Is the result numeric?
NO
9. Is the result= '<0.001'?
YES
10. Hold for manual review—Color Dark Purple—Add tech note "Below assay range—reanalyze"
If ({Result} {On Test} "TSH"= "<0.001") then {Hold Test for Verification} "TSH" {AND} {Set} {Test User Field 01} {On Test} {"TSH"} = "Below Assay Range–Reanalyze"
NO
11. Is the result= '>150.000'?
YES
12. Hold for manual review—Color Red—Add tech note "Above assay range—dilute and reanalyze"
If ({Result} {On Test} "TSH"= ">150.000") then {Hold Test for Verification} "TSH" {AND} {Set} {Test User Field 01} {On Test} {"TSH"} = "Above Assay Range–Dilute and Reanalyze"
YES
13. Round result to 3 decimal places
If ({Result} {On Test} "TSH" {Is Numeric}) then {Round}{Result}{On Test} "TSH" {To} "3" Decimal Places
14. Is result > 74.999?
YES
15. Hold for manual review—Color Pink—Add tech note "Above verification range Repeat testing"
If ({{Result} {On test} "TSH" {Is Numeric}) {AND} ({Result} {On Test} "TSH" > "74.999")) then {Hold Test for Verification} "TSH" {AND} {Set} {Test User Field 01} {On Test} {"TSH"} = "Above Verification Range—Repeat Testing"
NO
16. Is result < .005?
YES
17. Hold for manual review—Color Turquoise—Add tech note "Below verification range Repeat testing"
If ({{Result} {On test} "TSH" {Is Numeric}) {AND} ({Result} {On Test} "TSH" > "0.005")) then {Hold Test for Verification} "TSH" {AND} {Set} {Test User Field 01} {On Test} {"TSH"} = "Below Verification Range—Repeat Testing"
NO
No related assays should be evaluated with TSH
NO
No specimen integrity checks performed on this analyzer. Grossly hemolyzed, lipemic, or icteric samples should be addressed manually before analysis
18. Result is autoverified
Note:* Test User Field 01 issued as a note field in this particular example. It will be shown on the results screen.

Fig. 3. Example of an immunoassay reagent-specific algorithm

The analyzer-specific algorithm should be built in the same format as the master. The title should refer to the version of master from which it was derived and define the analyzer model with specific differentiating information. Each decision point and action should be numbered for easy reference once rules are created, and it is important to be diligent and disciplined with version control. See **Fig. 2** for an example of an analyzer-specific algorithm template.

In an optimal situation, analyzer purchase decisions would take the lab's autoverification algorithms into consideration. For example, it might be important to perform interassay comparisons, which can be performed adequately only when your AV tool receives all assays at the same time. Thus, it might be optimal to ensure the analyzer transmits all results at a single time on a particular patient. Interassay comparisons can be performed when results are transmitted one at a time, or even across analyzer platforms, but the risk of autoverifying a potentially inaccurate result increases because the initial result (that is part of the comparison) may be autoverified before a second (or third) result is received.

Reagent-specific algorithm. A reagent- (or test-) specific algorithm should be created only after the master and analyzer-specific algorithms have been defined. Not only must each measured test have its unique reagent-specific algorithm, but so too should each calculated test. All reagent-specific algorithms must refer to the correct version of the analyzer-specific algorithm from which it was derived. If the master and analyzer-specific templates are written appropriately, the reagent-specific algorithms become fairly easy to write; it is simply a matter of adding test-specific information to the analyzer algorithm. In this article I provide examples of reagent-specific algorithms for three laboratory disciplines (immunoassay, hematology, and chemistry). See **Fig. 3** for an example of a reagent-specific algorithm for TSH, based off the immunoassay algorithm in **Fig. 2**.

The master or analyzer-specific algorithms, or both, often contain decision points that are not relevant to a specific test. Any excluded decision points should be retained on the algorithm with a notation indicating it was skipped.

It is acceptable, and a good idea, to note the actual rules on the algorithm itself for the lab-specific AV tool as shown in the examples. Test the rules thoroughly to ensure proper functionality. Note the use of a variable field (labeled "Test User Field 01") to display notes to the reviewer. This field is specific to the tool used to create the rules and may not be available, but other fields could be used in its place.

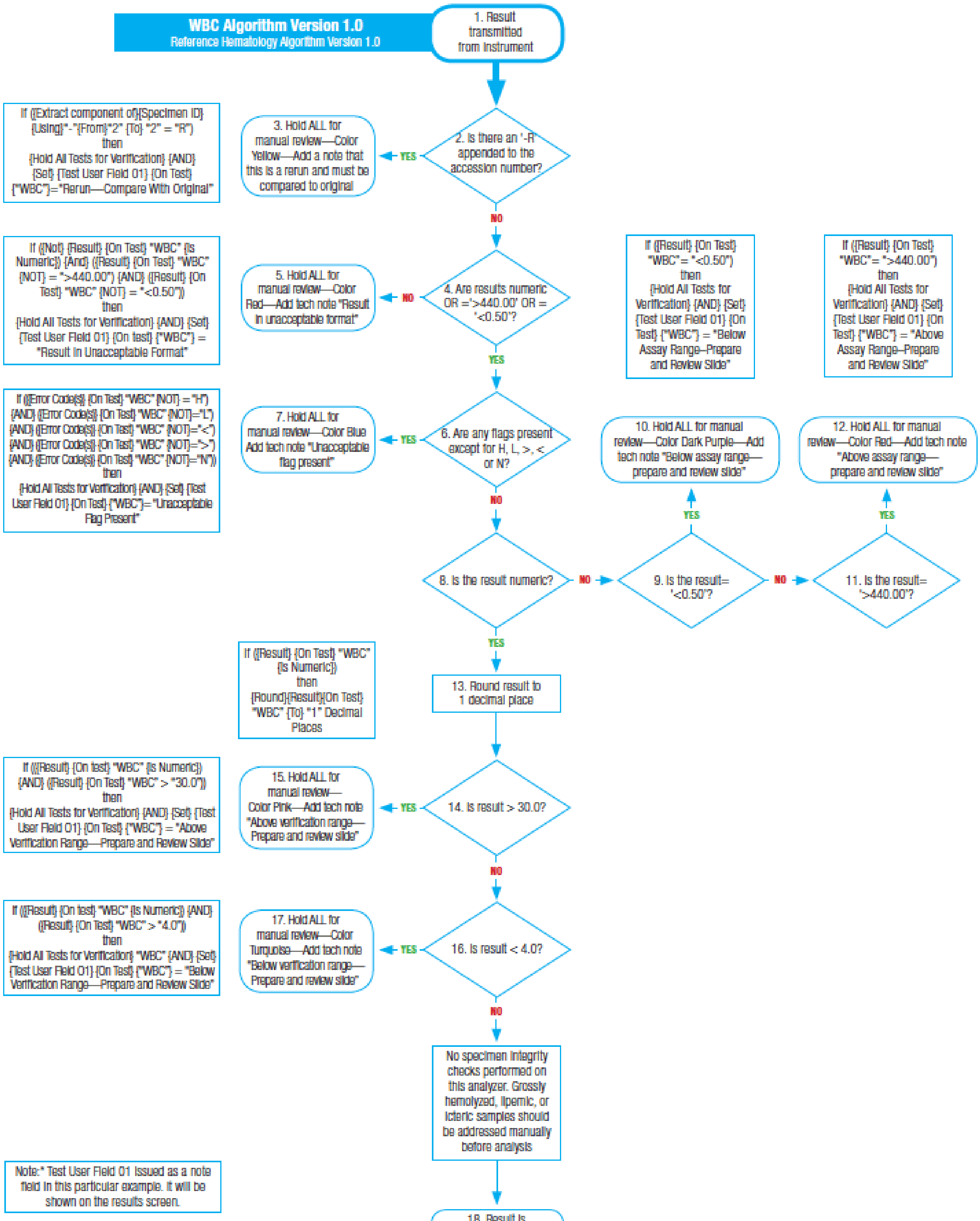

WBC Algorithm Version 1.0
Reference Hematology Algorithm Version 1.0
1. Result transmitted from Instrument
2. Is there an '-R' appended to the accession number?
YES
3. Hold ALL for manual review—Color Yellow—Add a note that this is a rerun and must be compared to original
If ({Extract component of}{Specimen ID} {Using}"-"{From}"2" {To} "2" = "R") then {Hold All Tests for Verification} {AND} {Set} {Test User Field 01} {On Test} {"WBC"}="Rerun—Compare With Original"
NO
4. Are results numeric OR ='>440.00' OR = '<0.50'?
NO
5. Hold ALL for manual review—Color Red—Add tech note "Result In unacceptable format"
If ({Not} {Result} {On Test} "WBC" {Is Numeric}) {And} ({Result} {On Test} "WBC" {NOT} = ">440.00") {AND} ({Result} {On Test} "WBC" {NOT} = "<0.50")) then {Hold All Tests for Verification} {AND} {Set} {Test User Field 01} {On test} {"WBC"} = "Result In Unacceptable Format"
YES
6. Are any flags present except for H, L, >, < or N?
YES
7. Hold ALL for manual review—Color Blue Add tech note "Unacceptable flag present"
If ({Error Code(s)} {On Test} "WBC" {NOT} = "H") {AND} ({Error Code(s)} {On Test} "WBC" {NOT}="L") {AND} ({Error Code(s)} {On Test} "WBC" {NOT}="<") {AND} ({Error Code(s)} {On Test} "WBC" {NOT}=">") {AND} ({Error Code(s)} {On Test} "WBC" {NOT}="N")) then {Hold All Tests for Verification} {AND} {Set} {Test User Field 01} {On Test} {"WBC"}="Unacceptable Flag Present"
NO
8. Is the result numeric?
NO
9. Is the result= '<0.50'?
YES
10. Hold ALL for manual review—Color Dark Purple—Add tech note "Below assay range—prepare and review slide"
If ({Result} {On Test} "WBC"= "<0.50") then {Hold All Tests for Verification} {AND} {Set} {Test User Field 01} {On Test} {"WBC"} = "Below Assay Range–Prepare and Review Slide"
NO
11. Is the result= '>440.00'?
YES
12. Hold ALL for manual review—Color Red—Add tech note "Above assay range—prepare and review slide"
If ({Result} {On Test} "WBC"= ">440.00") then {Hold All Tests for Verification} {AND} {Set} {Test User Field 01} {On Test} {"WBC"} = "Above Assay Range–Prepare and Review Slide"
YES
13. Round result to 1 decimal place
If ({Result} {On Test} "WBC" {Is Numeric}) then {Round}{Result}{On Test} "WBC" {To} "1" Decimal Places
14. Is result > 30.0?
YES
15. Hold ALL for manual review—Color Pink—Add tech note "Above verification range—Prepare and review slide"
If (({Result} {On test} "WBC" {Is Numeric}) {AND} ({Result} {On Test} "WBC" > "30.0")) then {Hold All Tests for Verification} {AND} {Set} {Test User Field 01} {On Test} {"WBC"} = "Above Verification Range—Prepare and Review Slide"
NO
16. Is result < 4.0?
YES
17. Hold ALL for manual review—Color Turquoise—Add tech note "Below verification range—Prepare and review slide"
If ({Result} {On test} "WBC" {Is Numeric}) {AND} ({Result} {On Test} "WBC" > "4.0")) then {Hold All Tests for Verification} "WBC" {AND} {Set} {Test User Field 01} {On Test} {"WBC"} = "Below Verification Range—Prepare and Review Slide"
NO
No specimen integrity checks performed on this analyzer. Grossly hemolyzed, lipemic, or icteric samples should be addressed manually before analysis
18. Result is autoverified
Note:* Test User Field 01 issued as a note field in this particular example. It will be shown on the results screen.

Fig. 4. Example of a hematology reagent-specific algorithm

be named as follows: GLU-RA2-S1- ple, changing results to positive or

Fig. 4 is an example of a reagent-specific algorithm for total white blood cell count that conforms to the ISLH (International Society for Laboratory Hematology) guidelines. Because most hematology results within a CBC are interrelated, in most cases it is appropriate to hold all tests received from the hematology instrument for manual review, as shown in the example.

Fig. 5 is an example of a chemistry reagent-specific algorithm. Chemistry algorithms tend to be more complex than those in hematology or other laboratory areas. The example (potassium) takes into consideration both serum index and interassay comparison rules.

Tool selection and implementation. It is common for laboratories to invest in information technology platforms before developing the requirements for their function. Process experts and consultants agree this is not good practice. It is only after the requirements are understood that a tool should be selected. AV tool selection should begin after analyzer-specific algorithms are developed. Some middleware and LIS vendors have "standard" rule packages, but it is important to understand the level to which these packages meet an individual laboratory's needs. A 60 percent AV rate can be achieved with some available tools, but a goal of 90 percent-plus is what the laboratory must attempt to attain.

The AV tool must be able to meet the needs of the laboratory as outlined in the algorithms. A single system for the entire laboratory reduces the cost of implementation, interfacing, and training, and the overall cost of service. It also allows cross-training of valuable skilled labor. However, the tool must be flexible enough to adapt to the specific needs of the individual departments. Implementation support, training, post-implementation support, clinical lab knowledge of the vendor, and cost must all be taken into consideration.

After selecting the proper AV tool, the laboratory and the vendor must define the parameters of implementation. Don't attempt to implement AV in all areas of the laboratory at once, but do ensure that the selected tool can be used when the time for implementation arrives. A scaled approach should be taken to minimize the impact on operations. The clinical, operational, and financial impact of AV will usually far outweigh the investment of incremental scaling.

Translation of algorithms into rules. One of the greatest difficulties of implementing AV is translating the algorithms into code that the tool can use to perform AV. Initially, the tool vendor should help the customer build the rules to match the reagent-specific algorithm. As stated, it is good practice to put the rule directly on the algorithm.

Naming the rules is also important, and I recommend doing so as follows: Test Name—Reagent Algorithm Version—Step Number. For reference, it might also be acceptable to add a brief description of the rule's function. For instance, a rule might be named as follows: GLU-RA2-S1-Check for rerun. Another benefit of naming the rules in this manner is that you can see (after the "S" for "Step") that the rules are in order because it may be important to ensure that rules are triggered for the AV system to function appropriately. Changing the version of a reagent-specific algorithm will require a review and retest of all previously written rules for that algorithm.

When writing rules based off of numeric results, it should be standard practice to first check if the result is in fact numeric. Some tools will evaluate non-numeric results as zeros, and thus could create significant problems, including autoverifying results that should not be autoverified. Another common pitfall is when rules change results to non-numeric values (for example, changing results to positive or negative as in toxicology). The rules triggered after the event could evaluate the non-numeric values as zero and cause a problem when troubleshooting.

Exceptions review process. One hundred percent autoverification is unachievable, and managing the AV exceptions (results that do not pass AV) can take a lot of time. Therefore, spend time understanding and optimizing the exceptions review process. Even when a laboratory isn't performing AV, an optimal review process with the proper information system can dramatically improve error rates, TAT, and FTE use. A perfect opportunity to implement a new exceptions review process is during AV implementation.

Arguably, the process and requirements for exceptions review can be defined before the tool is selected. Traditional LIS review screens offer no help to the user with the exception of a few error flags and perhaps a bold font. Tools today offer everything from colors and technologist-specific pop-ups to hematology images on the review screen.

Flexibility comes at a price, however, in that the laboratory must first define the requirements and specifications and only then can it implement the tool. The algorithms should identify how the exceptions appear on the screen and the colors or notes a user should see. A scheme to define the meaning of colors or fonts should be agreed on and standardized across the laboratory. For instance, red could indicate a critical value; yellow might indicate an unexpected error flag.

Substantiation. Testing and validating rules can be time-consuming, but it’s critical to test every rule each time an algorithm is updated. Some AV tools offer an added benefit of a testing area where rules can be tested without affecting production. However, initial validation should be done live and all documentation should mimic the documentation of actual patient samples.

After running quick tests on all rules to test functionality, document each test case and store the documentation electronically. A validation package should be provided with each analyzer specific for AV. The first page should be the AV SOP followed by the master algorithm and analyzer-specific algorithms. Each version that the individual reagent-specific algorithms reference should be present. Next, each reagent-specific algorithm should be followed by the test cases and evidence for each step in the algorithm. Each test case should include analyzer printout, audit trail (from AV tools), and patient report. A brief header page could include the step or steps of the algorithm to which the test case refers. The patient report should be from the final source of truth for the data (typically an electronic health record).

AV revalidation must be performed regularly, with the frequency dependent on the laboratory's regulating agency. The CAP currently requires a once per year revalidation of AV rules. Different AV tools have functionalities that may assist in these efforts and should be noted during tool selection.

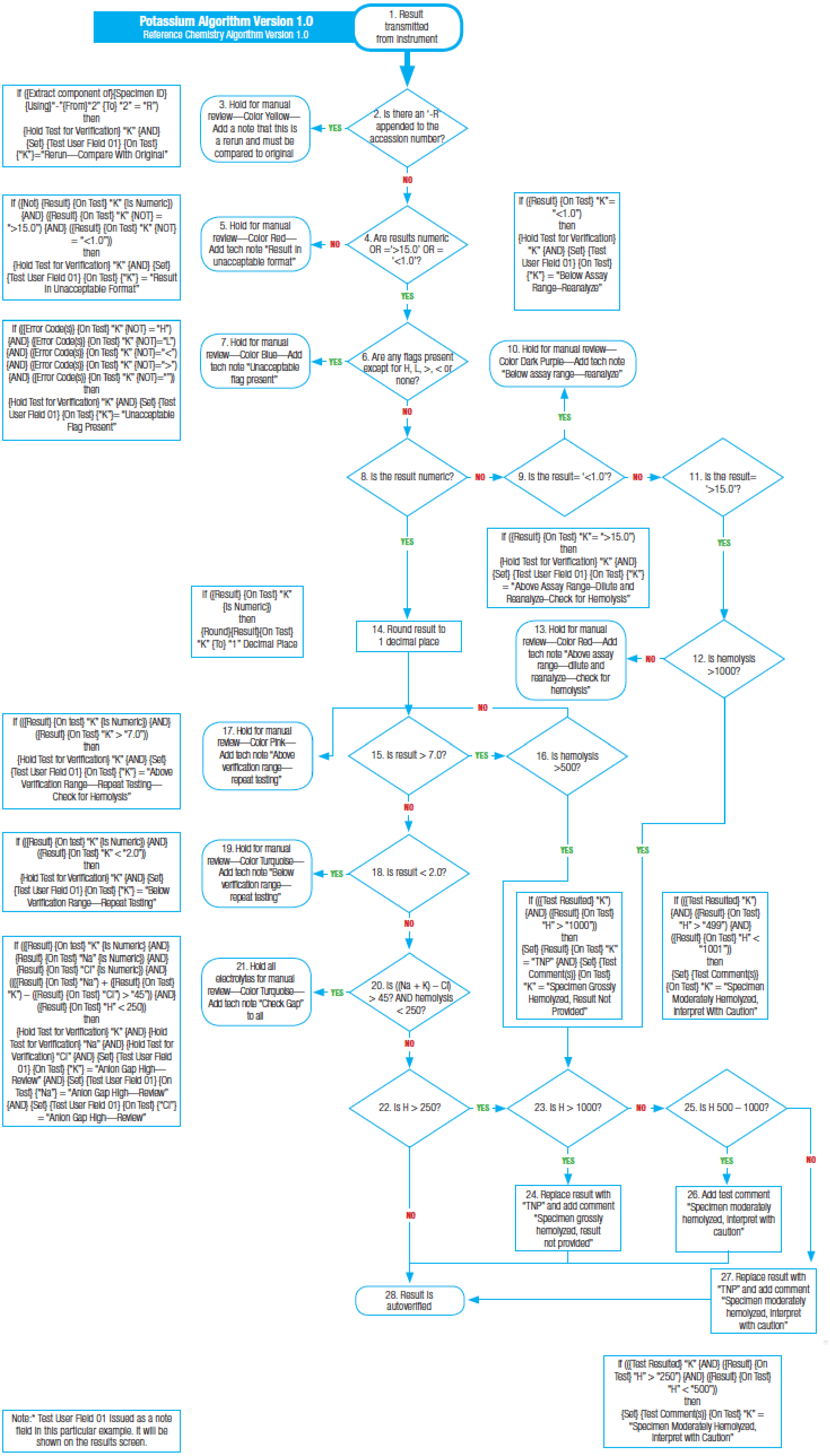

Potassium Algorithm Version 1.0
Reference Chemistry Algorithm Version 1.0
1. Result transmitted from instrument
2. Is there an '-R' appended to the accession number?
YES
3. Hold for manual review—Color Yellow—Add a note that this is a rerun and must be compared to original
If ([Extract component of]{Specimen ID} {Using}"-"{From}"2" {To} "2" = "R") then {Hold Test for Verification} "K" {AND} {Set} {Test User Field 01} {On Test} {"K"}="Rerun—Compare With Original"
NO
4. Are results numeric OR ='>15.0' OR = '<1.0'?
NO
5. Hold for manual review—Color Red—Add tech note "Result in unacceptable format"
If ({Not} {Result} {On Test} "K" {Is Numeric}) {AND} ({Result} {On Test} "K" {NOT} = ">15.0") {AND} ({Result} {On Test} "K" {NOT} = "<1.0")) then {Hold Test for Verification} "K" {AND} {Set} {Test User Field 01} {On Test} {"K"} = "Result In Unacceptable Format"
If ({Result} {On Test} "K"= "<1.0") then {Hold Test for Verification} "K" {AND} {Set} {Test User Field 01} {On Test} {"K"} = "Below Assay Range–Reanalyze"
YES
6. Are any flags present except for H, L, >, < or none?
YES
7. Hold for manual review—Color Blue—Add tech note "Unacceptable flag present"
If ([Error Code(s)} {On Test} "K" {NOT} = "H") {AND} ({Error Code(s)} {On Test} "K" {NOT}="L") {AND} ({Error Code(s)} {On Test} "K" {NOT}="<") {AND} ({Error Code(s)} {On Test} "K" {NOT}=">") {AND} ({Error Code(s)} {On Test} "K" {NOT}="")) then {Hold Test for Verification} "K" {AND} {Set} {Test User Field 01} {On Test} {"K"}= "Unacceptable Flag Present"
10. Hold for manual review—Color Dark Purple—Add tech note "Below assay range—reanalyze"
NO
YES
8. Is the result numeric?
NO
9. Is the result= '<1.0'?
NO
11. Is the result= '>15.0'?
YES
YES
If ({Result} {On Test} "K"= ">15.0") then {Hold Test for Verification} "K" {AND} {Set} {Test User Field 01} {On Test} {"K"} = "Above Assay Range–Dilute and Reanalyze–Check for Hemolysis"
If ({Result} {On Test} "K" {Is Numeric}) then {Round}{Result}{On Test} "K" {To} "1" Decimal Place
14. Round result to 1 decimal place
13. Hold for manual review—Color Red—Add tech note "Above assay range—dilute and reanalyze—check for hemolysis"
NO
12. Is hemolysis >1000?
NO
If ({[Result} {On test} "K" {Is Numeric}) {AND} ({Result} {On Test} "K" > "7.0")) then {Hold Test for Verification} "K" {AND} {Set} {Test User Field 01} {On Test} {"K"} = "Above Verification Range—Repeat Testing—Check for Hemolysis"
17. Hold for manual review—Color Pink—Add tech note "Above verification range—repeat testing"
15. Is result > 7.0?
YES
16. Is hemolysis >500?
NO
If ({Result} {On test} "K" {Is Numeric}) {AND} ({Result} {On Test} "K" < "2.0")) then {Hold Test for Verification} "K" {AND} {Set} {Test User Field 01} {On Test} {"K"} = "Below Verification Range—Repeat Testing"
19. Hold for manual review—Color Turquoise—Add tech note "Below verification range—repeat testing"
YES
18. Is result < 2.0?
YES
YES
If (([Test Resulted} "K") {AND} ({Result} {On Test} "H" > "1000")) then {Set} {Result} {On Test} "K" = "TNP" {AND} {Set} {Test Comment(s)} {On Test} "K" = "Specimen Grossly Hemolyzed, Result Not Provided"
If (([Test Resulted} "K") {AND} ({Result} {On Test} "H" > "499") {AND} ({Result} {On Test} "H" < "1001")) then {Set} {Test Comment(s)} {On Test} "K" = "Specimen Moderately Hemolyzed, Interpret With Caution"
NO
If ({[Result} {On test} "K" {Is Numeric} {AND} {Result} {On Test} "Na" {Is Numeric}) {AND} {Result} {On Test} "Cl" {Is Numeric}) {AND} (({[Result} {On Test} "Na") + ({Result} {On Test} "K") – ({Result} {On Test} "Cl") > "45")) {AND} ({Result} {On Test} "H" < 250)) then {Hold Test for Verification} "K" {AND} {Hold Test for Verification} "Na" {AND} {Hold Test for Verification} "Cl" {AND} {Set} {Test User Field 01} {On Test} {"K"} = "Anion Gap High—Review" {AND} {Set} {Test User Field 01} {On Test} {"Na"} = "Anion Gap High—Review" {AND} {Set} {Test User Field 01} {On Test} {"Cl"} = "Anion Gap High—Review"
21. Hold all electrolytes for manual review—Color Turquoise—Add tech note "Check Gap" to all
YES
20. Is ((Na + K) – Cl) > 45? AND hemolysis < 250?
NO
22. Is H > 250?
YES
23. Is H > 1000?
NO
25. Is H 500 – 1000?
YES
YES
NO
24. Replace result with "TNP" and add comment "Specimen grossly hemolyzed, result not provided"
26. Add test comment "Specimen moderately hemolyzed, interpret with caution"
NO
28. Result is autoverified
27. Replace result with "TNP" and add comment "Specimen moderately hemolyzed, interpret with caution"
If (([Test Resulted} "K" {AND} ({Result} {On Test} "H" > "250") {AND} ({Result} {On Test} "H" < "500")) then {Set} {Test Comment(s)} {On Test} "K" = "Specimen Moderately Hemolyzed, Interpret with Caution"
Note:* Test User Field 01 issued as a note field in this particular example. It will be shown on the results screen.

Fig. 5. Example of a chemistry reagent-specific algorithm

Train. All users of the AV system need not be trained to the same level, and different departments must be trained on different aspects of the AV system because requirements will vary. Continuous improvement will require ongoing training, and there should be an assignee designated to ensuring that training is a priority.

Scale. Any level of autoverification will result in significant process improvement. Once a single analyzer has been done up to about 60 percent, additional analyzers should be next in line. When the AV level is at 60 percent for all analyzers, the process should be started over to attain a higher level of AV. Regular reassessment of AV must be part of the laboratory's continuous improvement processes. The process improvement gained from implementing AV will make it possible for the laboratory to dedicate resources to AV. A balance must be maintained, however, because higher levels of autoverification become harder to achieve as the rates asymptotically approach 100 percent. The laboratory should always measure the return on investment when instituting AV improvement events, because it may not make sense to continue to improve an already efficient process.
[hr]

Bill Marquardt, C(ASCP), MBA, CSSBB, is vice president of laboratory intelligence/connectivity, Labsco, Louisville, Ky. Before he joined Labsco, he was director of IT services at BioReference Laboratories. He is chairholder-designate for the CLSI committee on department-specific autoverification guidelines.