CASE STUDY

BECS Selection – ISBT 128 Compatible or Compliant

Lynne Briggs – VP and CIO, Linda Gruber – Director Operations Quality , Petra Yurchich – Director IS Quality

Background:

In early 2016, the United States-based Versiti blood center (the affiliation of blood centers in Wisconsin, Illinois, Michigan, Indiana [with a recent extension into Ohio]), recognized, like many, that consolidating onto a single Blood Estalishment Computer System (BECS) and standardizing SOPs had many advantages, among them:

> De-risking the support of multiple and aging BECS across Versiti's operations;

> Lowering overall IT, as well as quality and general SOP and training costs; and

> Positioning Versiti with a combined donor base and manufacturing controls to optimize donor recruitment and operations across its geographies.

Process:

In seeking a single BECs, Versiti wanted to ensure it fully considered all the available FDA 510(k) cleared options, as well as consider more "out of the box" approaches such as writing its own BECS and/or partnering with another organization in creating or using their BECS. Suffice it to say, Versiti's desired timeline to reap the benefits of a single BECS did not allow for a lengthy development and/or higher-risk validation with a system which was brand new. Versiti's focus turned quickly to evaluating commercially available systems, but noting that some good work is being done in many institutions in developing their own BECS.

Moving forward, within the evaluation of those systems, the Versiti team felt that deep requirements analysis on basic operational functions and use of a BECS – including ISBT 128 compliance – was not where it would focus the bulk of its time. We would assume (but verify) that most BECS today dealt with the basics. Besides, Versiti had many unique requirements it wanted to ensure could be met.   These included use of multiple FDA licenses and time zones, mangement of high resolution HLA and antigen typings, the desire for well-integrated and robust electronic blood donation record features, which would also allow us to survey donors, allowing the provision of application program interfaces (APIs) to ease extensive integrations, and the ability to quickly respond to new technologies such as pathogen reduction. In short, the approach assumed that a 510(k), commercially available BECS, should meet all the basics (including ISBT 128), and we would spend more time focusing on the capabilties of a system needed for the unique aspects of Versiti and the hospitals and patients it serves.

What We Learned

While, in general, most of Versiti's assumptions regarding commercially available BECS were true, and we did eventually end up spending more time on our unique demands - we discovered or maybe "rediscoverd" two important aspects of evaluating any BECS:

1. True process and manufacturing controls are critical
2. Being ISBT 128 compliant is more than being "compatible with the bar code"

Briefly, relative to process controls/separation of duties – all BECS will record what you did, and keep an audit trail, etc. However, the ability of a system to provide for and manage critical control points for things like data entry, order of events, final verifications of items in a box for delivery to a hopsital, is important.   True manufacturing controls and check points differentiate a system and its ability to aid staff in avoiding mistakes. It's important to always verify how

the system will help the staff and/ or the holistic process of controls.   Its ometimes convenient for us to use or configure a system to "let us keep going", when, in fact, the controlled stop or check is highly beneficial.

More interesting was that nearly 10 years post the US adopting use of the ISBT 128 consensus standard, we still struggle a bit in understanding the importance of a system being truly ISBT 128 compliant.

     1.   A compliant system always makes use of the process control features of ISBT 128. For example, when manually entering a Donation Identification Number (DIN), the use of the check digit following the printed DIN (appearing inside a box) is essential. For mis-scans of the DIN, a check digit is incorporated into the Code 128 bar code itself. The use of these check characters creates a beautifully designed technology to ensure either a manual entry or a mis-scan is flagged if the corresponding check digit verification fails--getting the DIN right every time, all the time, regardless if the function is registering the donation or distributing the blood product.

     2.   A compliant system fully controls the creation and/or selection of the product code based upon all of the inputs up stream from bag type to manufacturing process and other attributes.   For example, you cannot request an irradiated red cell final label, unless you have confirmed the step of having irradiated the product. You cannot "pick" a product code from a drop down, without checks done against the bag type, aphersis equipment procedure, and being fully associated and checked against that product code. A well-designed and fully compliant system begins taking information associated with the DIN (such as bag type, maybe apheresis equipment, donation type) and calculates the final labeling.   Even if a product code had to be created ahead of ICCBBA providing one (research, etc.), the system should be able to be configured to allow for basic checks. Imagine a user being able to select a product code which does not reflect the bag type or anticoagulant used.

Summary/Conclusion:

Versiti was reminded that system complaince with ISBT 128 standards should never be taken for granted (as well as cGMP).   It's important to understand the rationale and intent of the various aspects of labeling, data, and systems standards.   They all work together to better ensure we have the right product, properly identified with all of its parts traceable, and that we have helped users avoid errors or mistakes. Simply being compatible with a bar code standard across systems or a data element was never the point….useful too, but not the point.   The point was built in controls.