**Protokoll Diskussion Dräxlmaier E3.HarnessAnalyzer Features 2020**

## Teilnehmer: Mildenberger, Aigner, Rappel, Rüsseler und noch 5 weitere Dräxlmaier Ma

## Ort: Ingolstadt

## Datum: 23.01.2020

## Potentialverteiler:

Der Zulieferer muss die Prüfung der Leitungen vom Potentialverteiler zum Gerät und für das dazugehörige Masse Netz durchführen und nachweisen. Dieser Schritt ist zeitaufwändig und basiert nur auf den KBL Daten. Ein Reengineering der Konnektivität muss über die Leitungssatzgrenzen hinweg durchgeführt werden. Diese Prüfung muss die Spannungsabfälle unter Berücksichtigung der Temperatur, des Leitungstyp-Spezifischen Widerstands, der Länge usw. ermitteln und ausgeben können. Es werden dabei zwei feste vorgegebene Konfiguration MIN/MAX vorgegeben.

* Die Eingangs-KBLs werden in einem Vorprozess mit Daten angereichert.
* Die nutzbaren Konfigurationen sind darin enthalten.
* Aus den Kundensachnummern der Leitungen kann der Leitungstyp ermittelt werden, man kann auch hier schon den spezifischen Widerstand und weiteren Informationen in Custom Properties in die KBL bringen.
* Stromwerte aus einer Datenbank werden an die Pins gebracht. (Man geht erstmal von Dauerströmen aus und in einem weiteren Schritt sollen auch Stromprofile hinterlegt werden können.)
* Temperaturen sollten an topologischen Segmenten der KBL hinterlegt werden können, aber vermutlich wird man auch noch Im HA für bestimmte Umfänge eine Eingabe schaffen müssen.

Wenn mehrere KBLs (xHCVs) geladen sind, müssen die Trennstellen aufgelöst werden (Namenskonvention). Nach manueller Auswahl eines Stromverteilers (über Namen erkennbar) sollen für diesen Stromverteiler die daran hängenden Verteilerbäume gefunden werden. Die Abgangspins können über Namenskonvention gefunden werden. Von dort aus ist der Pfad über Doppelanschläge und Splices bis zu den Komponenten zu verfolgen. Übergänge innerhalb von Komponenten werden nicht betrachtet. Von den Massebolzen aus (Namenskonvention der Stecker) werden die Massenetze rückwärts bis zu den Komponenten ermittelt. Wenn sich an einer Komponente genau ein Paar von Versorgung und Masse treffen, ist der Pfad klar. Wenn mehrere Massen an der Komponente enden, ist eine Meldung zu generieren. Wenn keine Masse da ist passiert dasselbe.

Die Versorgungsbäume sollen pro Pin dann komplett visualisiert werden (Verständnisprüfung) und für die einzelnen Leitungen ist der Spannungsabfall zu ermitteln. Die Temperaturen sind dann von den Segmenten auf die Leitungen zu übernehmen.

Die Änderung der Werte aus den KBLs muss nicht im HA ermöglicht werden.

Ein Export der Versorgungsbäume nach Excel ist nötig, evtl. wäre auch noch ein PDF-Export sinnvoll (der PDF-Export wäre als Endkunden (OEM) Export gedacht).

Nicht betrachtet werden heute Kontaktteile, Exklusivitäten und Innenbeschaltungen.

Alle Einheiten von Werten in Custom Properties werden festgelegt.

Die Namenskonventionen für diverse Erkennungen müssen konfigurierbar sein.

Für künftige weitere Betrachtungen ist zu überlegen, ob die entsprechenden Stromwerte, welche angereichert in der KBL enthalten sind, individuell zwischen zwei Ständen vergleichbar gemacht werden können.

Es kann sich für die Darstellung einiges vom Gesamt-Konnektivität-View abgeschaut werden. Zusätzlich kann hier ein Grid die Resultate pro Pin oder Stecker visualisieren, was dann als Basis für den Excel-Export dient.

Die Temperatur-Werte könnten auch in dieser View pro Draht angegeben werden.

**Zeitaufwandsschätzung: Implementierung 🡪 10 MW, Test 🡪 4 MW**

Screening-Liste:

Es gibt eine Liste, die die Kundenteilnummer für Kontakte mit einer Mindestleitungslänge in Verbindung bringt, die bei Ultraschall-Splices einzuhalten ist. Zu kurze Leitungen führen zu Kontaktbeschädigung. Die Funktion soll in einer KBL/HCV eine Prüfung auf Verletzung diese Bedingung durchführen und einen entsprechenden Report generieren. Dazu muss von jedem Ultraschallverbinder (Namenskonvention) die Leitungsbelegung angesehen werden und die Endterminals per Sachnummer in der Screening-Liste gefunden werden. Wird das Terminal nicht gefunden, ist eine Warnung zu generieren. Bei gefundenem Terminal kann die entsprechende Leitung gegen die definiert Mindestlänge geprüft werden. Die Länge soll für die definierten Längenklassen passieren, möglicherweise reicht aber auch die Verwendung der aktive Längenklasse aus. Module sind dabei nicht zu betrachten. Der entstehende Report muss noch detailliert werden, aber zumindest müssen neben der Leitungsnummer noch ein paar Angaben dazu kommen.

(Hier gab es noch Diskussionen, ob evtl. solche Screening-Informationen auch von einem Vorprozess in die KBL kommen sollten…)

**Zeitaufwandsschätzung: Implementierung 🡪 2 MW, Test 🡪 1 MW**

Excel Report der Datenvergleichsergebnisse:

Der heutige Excel Compare Result Report wird als zu groß und ungünstig strukturiert angesehen. Da im neuen HA die Möglichkeit besteht, einen Teil der zu vergleichenden Properties zu definieren, wird die Menge etwas geringer, aber die Ausgabe von Referenz und Vergleichswert im Kommentarfeld ist so nicht brauchbar. Hier will man die Werte in richtigen Zellen haben, damit man auch mit Filtern und Makros arbeiten kann. Bei hierarchischen Strukturen wird er Export zu unübersichtlich, flache Strukturen sind gewünscht.

Wir bekommen Beispiele für typische Reports, die aus Sicht von Dräxlmaier sinnvoll sind. Ideal wäre eine Konfigurationsmöglichkeit – aus Aufwandsgründen vermutlich aber nicht realisierbar.

**Zeitaufwandsschätzung: Implementierung 🡪 2 MW, Test 🡪 1 MW**

## Weiteres Vorgehen:

Dräxlmaier wird die nötigen Properties für den Verteilerbaum zusammenstellen und den Ort der Einpflege definieren. Wir schätzen den Aufwand für diese Funktionalität ab und machen eine Skizze/ Vorschlag, wie das im HA aussehen könnte.

Wir bekommen Daten für eine Screening-Liste, es gibt dazu aber kein fest definiertes Format, wir können auch Vorschläge von unserer Seite machen.

Wir bekommen einige Report Samples.

Der Wunsch besteht, vor allem die Funktion Versorgungsbaum in einem nächsten Release nutzen zu können. Grundsätzlich ist aber auch eine prototypische Implementierung erstmal denkbar.