Intelligent Part Numbering Systems for Off-The-Shelf Components – Why?

A DCA White Paper - posted 6/6/2002

Intelligent part numbering schemes for electronic components (where "intelligent" means that the internal part number assigned carries information about the part) have been used for decades. This paper will explore why these past approaches are crippling today's collaborative product commerce efforts, and will describe better approaches that can save your company significant dollars.

Intelligent Part Numbers (IPNs) are used as a mnemonic to identify components and other items when database storage and searches are expensive or difficult, or systems fail to contain sufficient information about a part for some purpose, such as identifying the right buyer in an MRP system.

Intelligent schemes for components are typically based on the part type and typically are implemented, as a minimum, in a part number prefix, e.g., if "03" is defined as the prefix for memory devices, then 03-123456 is a memory device. The prefix is normally associated with a component class that is defined by the OEM (e.g., capacitor, resistor, cable, oscillator, logic IC, BOM, Schematic, and so on). OEMs that use a prefix in this manner risk running into trouble. Most problems are caused by inadequate scoping, resulting in too many component functions under a single prefix. Others are related to inaccurate prefixing (e.g., giving an analog part a logic prefix), insufficient headroom (too few part numbers available for a commodity), or excessive complexity. We've always been able to identify current and future problems with every implementation we've seen.

Sometimes, even the main part of the part number ("123456" in the example above) contains intelligence. Schemes exist which encode supplier name, part configuration, and other information in to the number; every numeral has meaning! This is overwhelming to the new or casual user who must reference a part number.

If a part number has a suffix, it usually indicates the revision level of the part. This is important for Bills-Of-Material (BOMs) and custom parts since it reflects (usually significant) changes to the product and documentation. Off-the-shelf component parts typically do not require a revision level. Indeed, maintaining revision levels for components would require a change to every single BOM a component is associated with when a manufacturer changes their part number, or it becomes obsolete, or a new component is added to an AML (Approved Manufacturer List; also know, less accurately, as AVL, or Approved Vendor List). Revisions are not necessary for off-the-shelf components because the internal part number represents a requirement, not a specific component manufacturer's part. Parts can change; the requirement cannot.

Intelligent part numbering schemes are costly. They require manual setup, maintenance, and often manual part number creation and entry due to their opaqueness, complexity, and poor support in off-the-shelf Component Management systems. By definition, this requires human intervention and, therefore, introduces cost, delay and the significant potential for inaccuracy into the part number assignment process. Error rates for manually assigned intelligent part numbers typically run in the five to ten percent range.

Today you can (or should be able to) store extensive parametric, classification, and other information about parts that is searchable and reportable. So where does the value in a complex, manually assigned part numbering system lie? That's the key question to answer to justify an intelligent part numbering system.

Some Product Data Management (PDM) or Collaborative Product Commerce (CPC) systems seem to drive OEMs towards intelligent part number strategies. In addition, many people are just used to and comfortable with intelligent numbers and want to keep using them. Moreover, many of today's commercial systems have poor searchability and reporting capabilities.

Fundamentally, the problem intelligent part numbering schemes seek to address is searchability. If a tool does not implement a well thought out classification schema, capture useful parametric values for components, or provide a standard for searchable part descriptions, users believe their sole recourse is an intelligent part numbering scheme. This is NOT the only, or best, answer!

DCA firmly believes that intelligent part numbering systems should have no place in today's OEMs. Every OEM seems to need to come up with their own part numbering system and, in doing so, reinvents the wheel. They have neither the time nor the "big picture" knowledge to do the job completely, accurately, and correctly.

Surprisingly, the tools and systems that OEMs use fail, almost universally, to provide more than a platform for implementing a simple component search. Our criteria for out-of-the-box functionality for tools includes:

- A robust, industry-standard classification schema
- Hierarchical sub-classes, up to five levels deep
- A standard, but expandable and modifiable, set of component parameters mapped to the classification schema and inherited through the tree structure
- A standard, configurable, part description that is class-specific and automatically generated
- Auto-assigned internal part numbers

Great, you say. Where do you get all this stuff? The starting point is clearly a "robust, industry-standard classification schema" for electronic components. While somewhat of a "holy grail", there are several classification schemes published (e.g., IEC, ECALS, RNTD). We recommend reviewing the UNSPSC. The UNSPSC™ (http://www.unspsc.com) is the "United Nations Standard Products and Services Code". While intended to classify everything from goats to DRAMs, there are only a few segments of interest to OEMs in the High Tech/Electronics area. These include Segments 31 (nuts/bolts), 32 (electronic components), 40 (fans/blowers), and 43 (computer equipment and software).

Today, Segment 32, perhaps the most critical, is weak, inaccurate, and incomplete for today's OEM. However, it's currently under revision. The revision is quite dramatic, which is necessary, and is being driven by RosettaNet (www.rosettanet.org), although it will not directly reflect the RosettaNet Technical Dictionary (RNTD) classification structure (or lack thereof). Changes are to be expected as well in Segment 43. For instance, subassemblies currently are handled inadequately in UNSPSC; some are in Segment 32 (e.g. 32101504, Surface mount circuit assemblies), while others are in Segment 43 (e.g., 43172005, Graphic accelerator cards). The current vision of the Segment 32 revision team is that a new segment for subassemblies is required for the UNSPSC to be successful. Contact DCA or the UNSPSC for status on this important standards effort. If you have expertise and some time, join the effort. The more industry effort is put into making this a useable, useful standard the less effort every OEM will have to expend to define their own schema.

UNSPSC, in addition to a four level classification schema, assigns eight digit numbers to every line item at every level. It can therefore replace internally developed "Material Codes" or "Class Codes". If every downstream system (ERP, MRP, etc.) uses a common, standard class code, electronic data mapping tasks will become much easier, especially company-to-company.

OK. Once we have a decent classification schema, where do we get the parametrics? How do we standardize on them when every component manufacturer wants to define the same parameter differently in order to either differentiate themselves or obscure compatibility? This is the task of the RosettaNet Technical Dictionary and, while the RNTD project is moving slowly, it's already developed a fair list of parameters ("properties" in RNTD-speak) for some component types. Once you have the parametrics, how do you maintain them? That's the subject, perhaps, for another editorial! In the meantime, pick the top five or so parameters for your needs in each class.

Once you have the parameters, a combination of static (e.g., CAPACITOR) and dynamically assigned values (e.g., X7R, 0.01µF, 50V, 0805) can be concatenated and assigned to the part description field of the AML. This provides a hint about which parameters to select for each class.

At that point, components can be found by manufacturer name, part number, description search, or parametric search (assuming the search capabilities of the tool are sufficient). Components on a BOM can be sorted as well by class to group all capacitors together, all memories together, etc. again assuming the tool has these rather straightforward capabilities. Reports can be provided by manufacturer, by part class, and so on.

So, in this perfect world, why would you need intelligent part numbers? Tell us why or provide other thoughts on the subject at info@designchainassociates.com.

