The European Network for B2B Integration in High Tech Industries

EDIFICE Set Label Guideline

Identification of Hierarchical Structured Product Data on Product Packages

Issue 1
Endorsed on 15 June 2011

EDIFICE
The European Network for B2B Integration in High Tech Industries
EDIFICE secretariat
Dora Cresens
Tiensestraat 12
B-3320 Hoegaarden
Belgium
Tel: +32 16 437 415
Email: Dora.Cresens@edifice.org

Publication Summary

Title:	Set Label - Identification of Hierarchical Structured Product Data on Product Packages
Author (s):	EDIFICE/ADC
Issue number:	Issue 1
Date of Issue:	15/06/2011
Num. of Pages:	36
Readership:	All
Language:	English
Abstract:	In the IT and telecommunications industry's supply chain, packages may contain many products. These products often consist of components which may also be relevant for tracking & tracing (e.g. bundles or modules/ boards as preconfigured "assembled products"). Automatic data capture infrastructure already in place normally is not used for the capturing of those hierarchical structures as bar codes mostly are used to identify the type and instance of the bundle/assembled product. This guideline defines how to encode multiple product information into a high capacity media (e.g. 2D symbols or RFID tags) in a hierarchical and structured way. It enables tracking of assets on a type and instance level even for hierarchically structured products while fully relying on established ISO/ ANSI standards.
Comment:	Comments and change requests to this document should be submitted to: EDIFICE secretariat
References:	ANS MH10.8.2 – Data Identifier and Application Identifier Standard EDIFICE Guideline – Product Package Label EDIFICE Shipment Label Guideline Issue 5 EuroData Council PaperEDI The „EDI-Mark"E.D.C. issue rev. 090924/100222-E ISO/IEC 16022 Data Matrix ISO/IEC 15415, Information technology — Automatic identification and data capture techniques — Bar code print quality test specification — Two-dimensional symbols ISO/IEC 15418 – Information technology — Automatic identification and data capture techniques — Data Identifier and Application Identifier Standard

ISO/IEC 15434 - Information technology — Syntax for high capacity ADC media

ISO/IEC 15459, Information technology — Automatic identification and data capture techniques — Unique identification – contains 6 parts. The parts related to this guideline are:

ISO/IEC 15459-2 – Information technology — Automatic identification and data capture techniques — Unique identification —Part 2: Registration Procedures

ISO/IEC 15459-3 – Information technology — Automatic identification and data capture techniques — Unique identification — Part 3: Common rules

ISO/IEC 15459-4 – Information technology — Automatic identification and data capture techniques — Unique identification —Part 4: Individual products and product packages

ISO 22742 Packaging - Linear bar code and two-dimensional symbols for product packaging

Table of Contents

List of Figures

List of Tables

Comparison to previous issue

No previous issue exists

1 Introduction

Transparency of the supply chain helps to minimize capital and operational expenditure by enabling the business to know where purchased assets are within the supply chain. Depending on the level of detail used with the definition of assets, it is important to know the details of products and components not only on a type level but also on an instance level. To minimize uncertainty on where assets are within the supply chain, material information should be captured as soon as it enters the company's goods receipt and should be fed into the relevant IT Systems. This also helps when taking inventory into account for goods still on the road or on the installation site and reduces undesired loss of materials.
Most businesses therefore already employ automatic data capture (ADC) technologies and count on standardized processes and structures. When capturing product and instance data, usually automatic capturing technologies like bar codes, 2D symbols and/ or RFID are used.

In the IT and telecommunications industry packages may contain many products. These products[1] often consist of components which may also be relevant for tracking & tracing. As of the development of this guideline, ADC infrastructure already in place normally is not used for the capturing of those hierarchical structures as bar codes mostly are used to identify the type and instance of the bundle/assembled product. Transmission of the data is instead done via electronic data exchange and/or printed documents sent with (attached to) the product package. Additionally transmitting the information about the content of packages as part of the package marking would enable identification of the content of the package using ADC technologies, as a complement to existing solutions (i.e. EDI and paper).

To enable fast capture of the type and instance identification of a hierarchical structured product without opening the package or correlating electronically received data, this document defines how to encode multiple product information in a high capacity media (e.g. 2D symbols or RFID tags), in a hierarchical and structured way. The capability of automatic identification of the content in a package supports the goods receiving, warehousing, inventory management and redistribution processes. Using this specification, the Lifecycle Management can be extended from products down to components of products. The components can be imported into relevant systems and tracked on a type and instance level.

The labeling guideline primarily refers to the labeling of product packages. For transport and shipment labeling see EDIFICE Shipment Label Guideline Issue 5.

Overview of the document:
Section 2 provides the scope of the hierarchical code usage as defined in this document. Section 3 presents different scenarios of product configurations that would benefit from the hierarchical structure. Section 4 specifies the rules for encoding in detail. It shows which data identifiers have to be used and how to build up the resulting data string. Section 5 gives recommendations for the application of the guideline. Section 6 provides examples for the usage of hierarchical structures and section 7 lists hints regarding data carriers.

[1] For example such products could be preconfigured products (e.g. routers containing modules) which will be placed into a rack/ cabinet. This guideline relates to the physical content of the package and not to the abstract representation of a structure in an IT System.

2 Scope

The usage of a hierarchical code specification is recommended for product packages which contain multiple products, e.g.:

- One or multiple hierarchical structured products (assembled from different components/products)
- Multiple products (without listing components)
- Application focus is to support inventory management, tracking and tracing throughout the supply chain

This guideline does not:

- Define how to set up the package/product structure. This is defined by the supplier
- Define the elements or levels of a product structure to be encoded. It is up to mutual agreement between supplier and customer to define the minimum, with the supplier having the option to add other elements/levels
- Cover if all products/components in a BOM (Bill of Material) for an assembled product are to be included in a hierarchical structure bar code. What to be included could be based on:
    - Serialization, i.e. only products/components assigned a serial number are to be included
    - Highest level and included Field Replaceable Units (FRU)
    - What to be included in a hierarchical structure is defined in a mutual agreement between trading partners (i.e. manufacturer/supplier and customer)
- Address marking below product package level. Please refer to section 5 for further guidance

2.1 Terms

For the purpose of this guideline the following terms and definitions apply:

BOM (Bill of Material)
List of the raw materials, sub-assemblies, intermediate assemblies, sub-components, components, parts and the quantities of each needed to manufacture an end product

Component
Item on the lowest hierarchy level and not assembled of other components

DSLAM
A Digital Subscriber Line Access Multiplexer is a network device that connects multiple customer Digital Subscriber Lines (DSLs) to a high-speed Internet backbone

Configured/assembled product
A product consisting of other products/components

Hierarchical structure
Representation of the physical structure in the barcode

Product
Products themselves can contain other products. A product that is contained in an configured/assembled product as a component also could be sold on its own as a product. In the same way, a product which is sold on its own also can be part of a configured/assembled product and therefore be seen as a component.

Product package
Packaging including one or more products/components and/or other packages

3 Hierarchical Structures and Application Scenarios

3.1 Overview

Several products, which themselves are possibly hierarchically structured can be contained in one product package.
Figure 1 shows the possible scenarios of content that a hierarchically structured symbol can contain. Horizontally, different products can occur, while vertically the products can be composed of other components and/ or products.

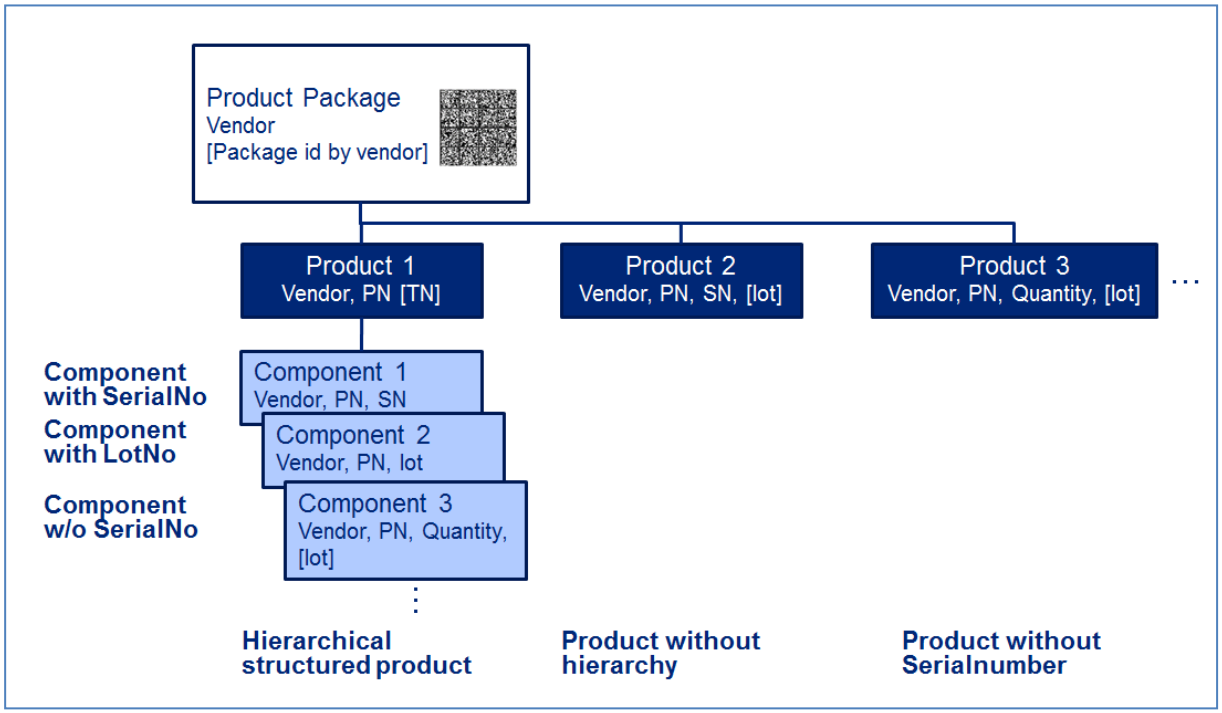


Figure 1 Hierarchical Structured Product Data

Notes:

- *The full complexity as shown in Figure 1 normally would not have to be utilized. Instead in most cases the package typically contains either a structured product, multiple structured products, or multiple non-structured products. Therefore the two scenarios described in the following chapters are discussed and defined in detail*
- *The current release of this guideline focuses on component, product and product package level. Higher logistics levels like pallets or transport packages might be added in a future release*

3.2 Scenario "A": Hierarchical structured products/products assembled from different products/components

In some industries like the telecommunications industry it is common to form preconfigured products/bundles of hardware and software. Besides being sold as a bundle, the products/components itself might be sold separately on their own and have their own product life cycle. The pre integration/configuration of the hardware components eases assembly for the customer. Purchasing and logistics processes for both customer and vendor are simplified. In order to identify a specific configuration/bundle, a dedicated part number is assigned to the bundle. The products/components each carry their own part numbers which have to differ from the part number of the assembled product.

When built up/installed, the assembled product and its part number could lose its relevance as potentially every product/component could be replaced later on by adding or removing products/components (i.e. the configuration therefore changes and the data initially related to this configuration instantly lose validity).

Figure 2 shows an example of an assembled product using several products/components which also can be sold independently (e.g. as spare parts or upgrade parts).

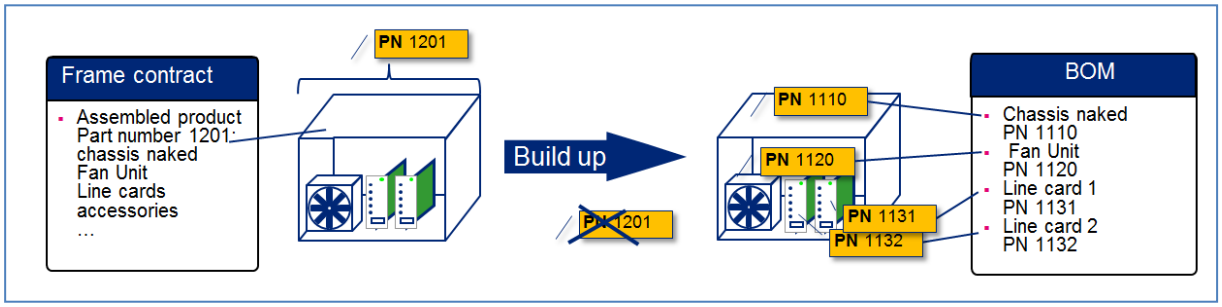


Figure 2 Example of an assembled product

Figure 3 illustrates the conceptual possibilities of hierarchical structured products regarding assembled products (one assembled product in a product package on the left hand side and two assembled products on the right hand side).

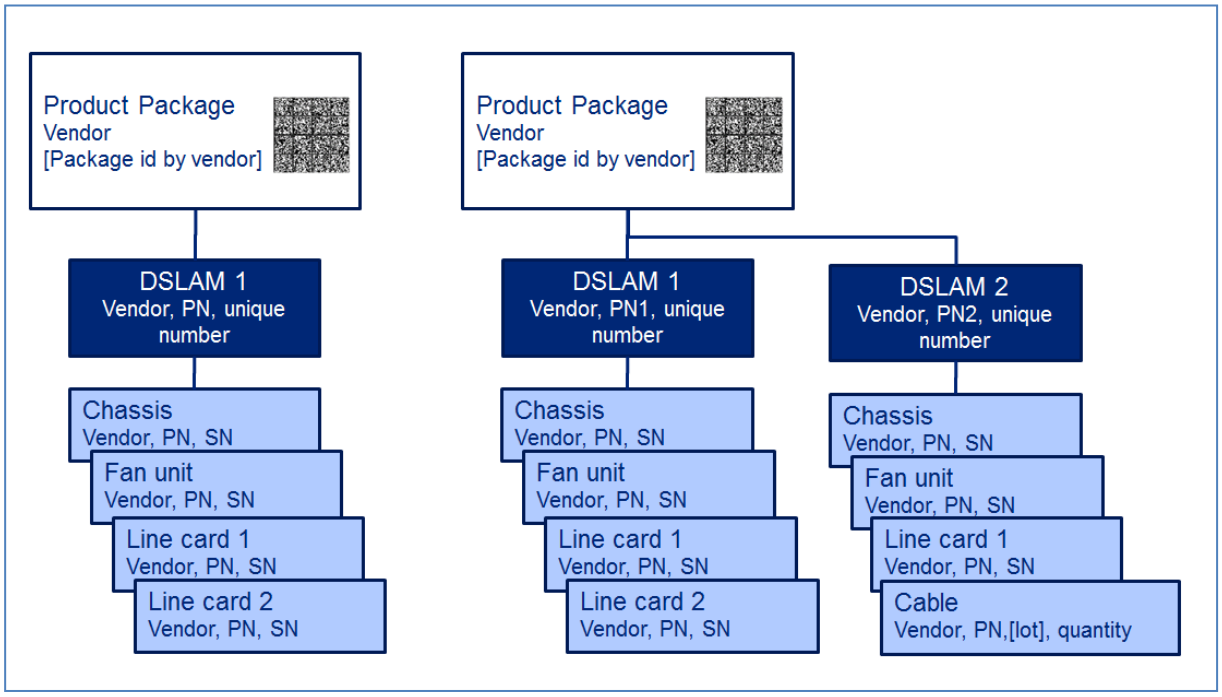


Figure 3 Example assembled products with hierarchical structure

Figure 4 combines the conceptual representation of a hierarchical structured product with a concrete assembled product.

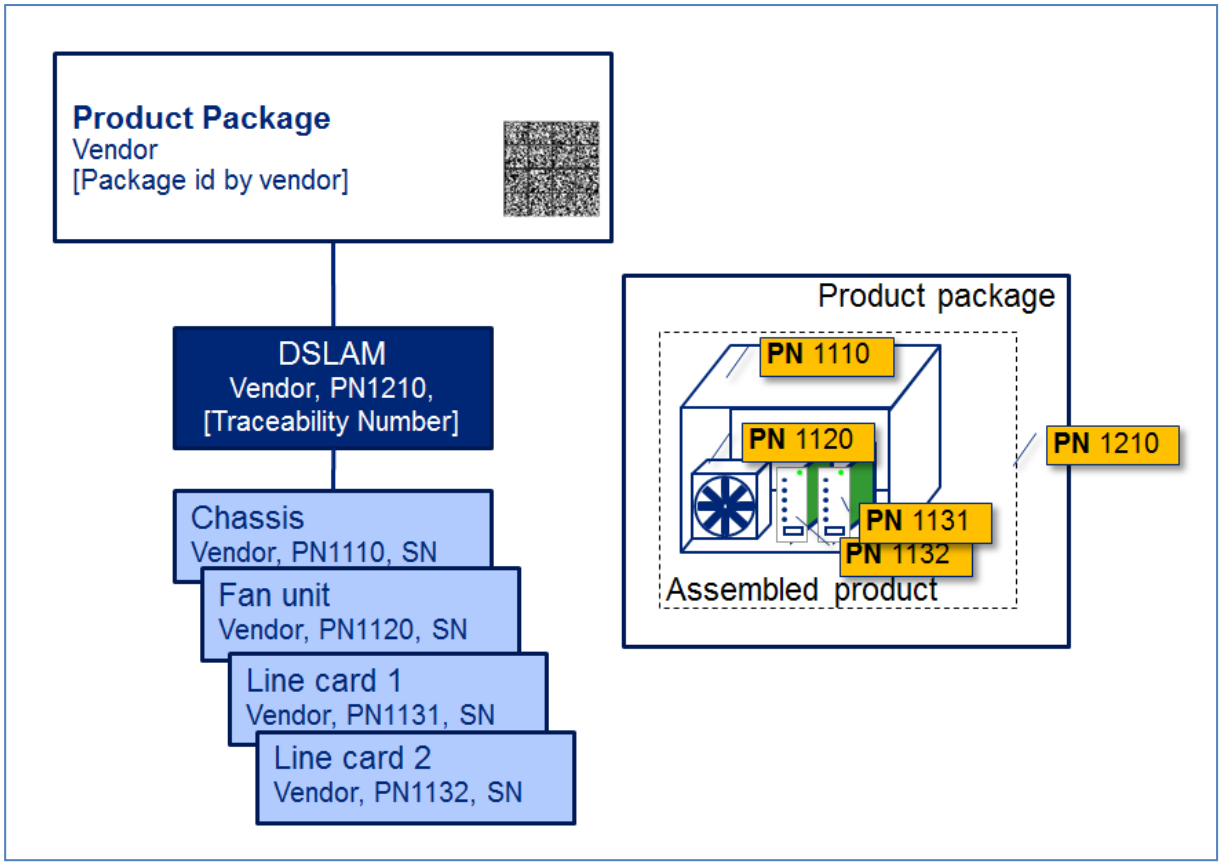


Figure 4 Conceptual representation of an assembled product

Notes:

- *Instead of being made up only of products/components, assembled products might contain other products which are made up of products/components themselves. This leads to further hierarchy levels. In this guideline it is assumed that the component is used as the lowest level of traceable items*

- *When having only one product in the package, the product package level might seem to have limited added value. So if the package contains only one product and the package does not have a package number, it is allowed to have the product as the top level element.*
  If the product is the top level it has to have a unique number.
  Anyway this is not recommended as the life of the information is only valid as long as the package is not opened and the product is not taken out of the package.
  Consequently first priority is to always have a product package level and omit the package level only if really necessary

- *Products themselves can contain other products. This does not change the product itself but the hierarchy level code to be used (HLC, also see Table 6)*

3.3 Scenario "B": Multiple products without components within one package

Figure 5 Multiple products without hierarchy within one package, describes several products that are sold as one product, e.g. a mobile phone together with a manual, headphones and a charger. In this example the customer buys a mobile phone, but expects to receive the complete package. A 2D symbol code could allow a cross check against order (or invoice, delivery papers…) whether all components are present in the product package without unpacking the package.

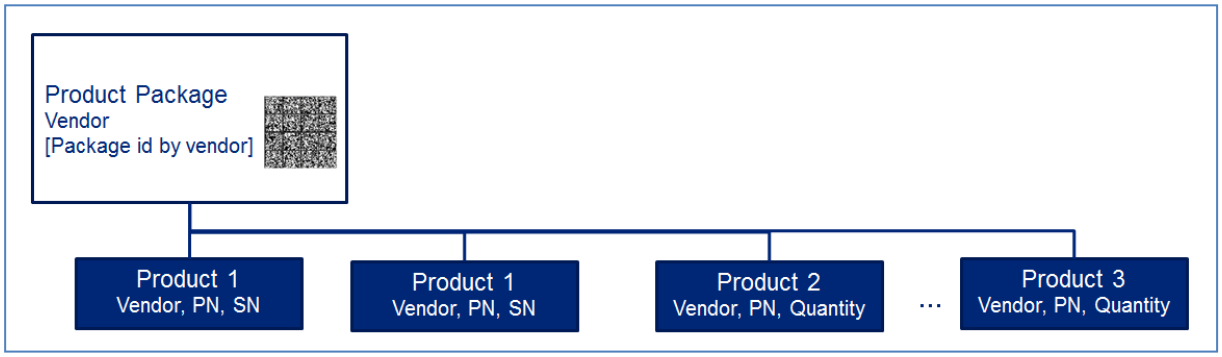


Figure 5 Multiple products without hierarchy within one package

Note: Having only one product which is not hierarchically structured, it is not appropriate to use a hierarchical structured symbol, as there is no hierarchy. If the hierarchical structured representation is used anyway in a 2D symbol, it should be used in the same way as if there was more than one product within the package

Sub configurations of scenario "B" encompass:

- One product without hierarchy
- Multiple Products of the same type within one package
- Multiple Products of different types within one package

4 Data encoding

Coding of product information is generally based on a three-layer model. The physical layer is the medium which carries the information. This could be a label on which a bar code and/or 2D symbol is used or an RFID tag. The syntax layer determines the structure of the information on the physical layer. The semantic layer gives meaning to the elements of the syntax structure by the usage of data identifiers. Figure 6 provides an example of the three layers for a two dimensional barcode.

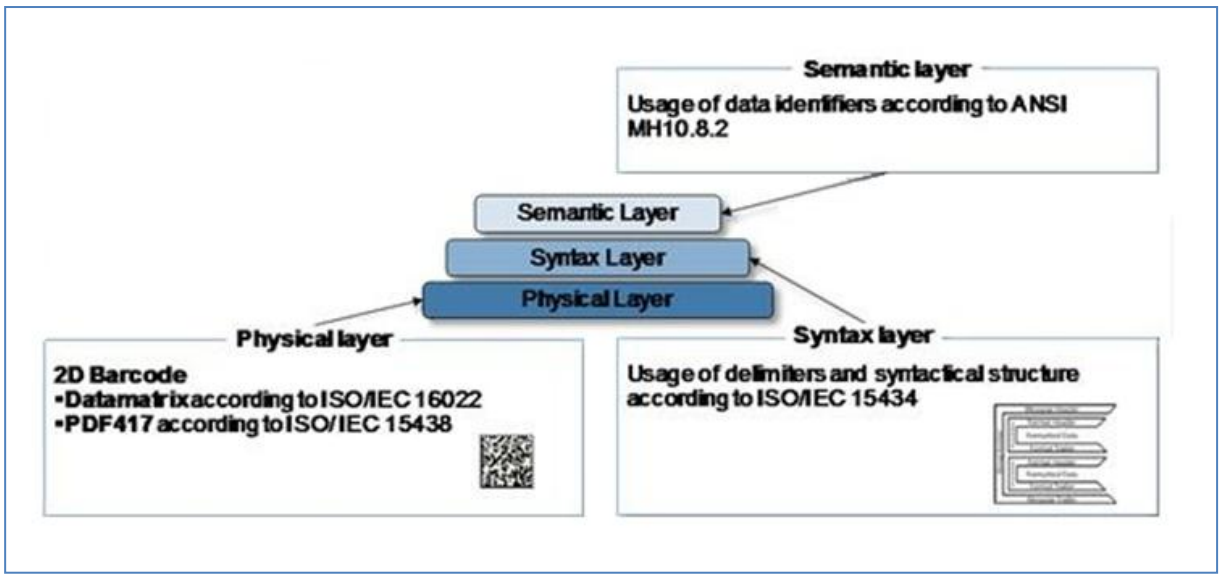


Figure 6 Overview on 2D Symbol specification domains

4.1 Physical layer/ data carrier/ medium

The physical layer is the transport medium of the coded information. For product identification the most common transport medium 2D symbols are:

- Data Matrix ISO/ IEC 16022
- PDF417 ISO/ IEC 15438
- QR Code ISO/ IEC 18004

The same information can be handled by using RFID technology. In this case agreements have to be met on compatibility of components, standardization (e.g. writing/reading, frequency, memory size, user memory usage, etc.) and choice of transponders.

For more detail on data carriers see Appendix A - recommendations on data carriers.

4.2 Syntax layer

The syntax for High Capacity Media as defined by ISO/IEC 15434 has been designed to allow encoding of a large amount of data. The structure fits in 2D symbols or RFID tags. As many vendors already mark their products using ISO/IEC 15434 labels in conjunction with Data Identifiers (DIs), this is the most suitable structure for the marking of hierarchical structure on packages.
The syntax contains a ”Message Header“ followed by the ”Format Header“ for identifying the embedded data structure (06 denotes the DI structure, Application Identifiers (AI) would be referenced by ID 05, UN EDIFACT by 04, etc.). The format ID will be translated at the scanning point either by intelligent scanners, by a data entry module or by the relevant IT system to derive the correct syntax elements like separator characters etc. The separators to be used with the DI structure are low value, non-human readable ASCII characters which will be displayed as e.g. “$^{R}_{S}$” “$^{G}_{S}$” in this document. A group

separator ("$^{G}_{S}$") is placed between single data elements that are preceded by a relevant DI. After the concatenated data elements, "Format Trailer" ("$^{R}_{S}$") and "End of Message" "$^{E}_{OT}$" characters will terminate the message. A standard syntax shall be illustrated below:

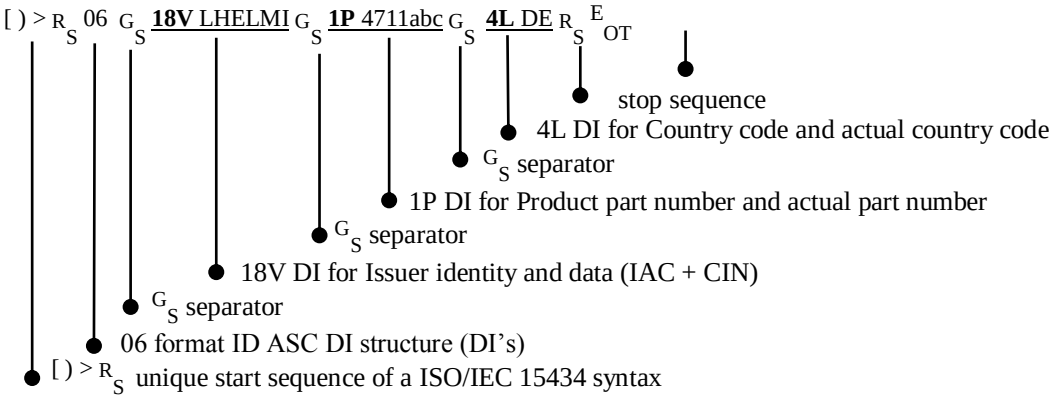
Start sequence: [ ) > $^{R}_{S}$
Format ID: 06
Separator: $^{G}_{S}$
Data element preceded by a Data Identifier (DI)
Separator $^{G}_{S}$
Next data element with DI and Data
...
Stop sequence $^{R}_{S}$ $^{E}_{OT}$.

A typical data string would look as follows:

[ ) > $^{R}_{S}$ 06 $^{G}_{S}$ DI DATA $^{G}_{S}$ DI DATA $^{G}_{S}$ DI DATA $^{G}_{S}$ ..... $^{G}_{S}$ DI DATA $^{R}_{S}$ $^{E}_{OT}$

Note: Blanks are inserted for illustration purposes but real data strings do not include blanks.

Sample of a data string filled with data:



4.3 Semantic layer

To give a meaning to the coded information data identifiers are used. The data elements to be allowed are defined by the American National Standards Institute (ANSI) in MH 10.8.2 (also identifiable as ISO/IEC 15418). It defines a list of data identifiers which commonly are used with the syntax as defined in ISO/ IEC 15434.
For the marking of hierarchical structured products, especially the data identifiers/data elements as described in the following chapter 4.3.1 are to be used.
To allow for hierarchies within the high capacity symbol, the data identifier "F" is used which is described in more detail in chapter 0.

For further detail also refer to ANSI MH 10.8.2 and the continuously maintained document.

4.3.1 Description of MH 10.8.2 Data Identifiers (DIs)

The following table lists the data elements which should be used to provide the necessary data for proper identification of hierarchical structured products.

F	Looping Header as defined as Section VI of MH 10.8.2 (also see chapter 0 of this document)
nJ	Unique license plate number (choice of available J DIs, can be used additionally or as substitute for 3S)
4L	Country Code for Country of Origin (“Made in”)
1P	Item Identification Code assigned by Supplier
25P	Identification of a party to a transaction as identified in 18V, followed by the supplier assigned part number
Q	Quantity, Number of Pieces, or Amount (numeric only) (unit of measure and significance mutually defined)
13Q	# of # ("this is the nth piece of x pieces in this shipment") Presented in the format "n/x", where the "/" (slash) is used as a delimiter between two values. See Annex C.6.3 of ANSI MH 10.8.2 for further information
S	Serial number or code assigned by the Supplier to an individual entity for its lifetime (e.g. computer serial number, individual traceability number)
3S	Unique Package Identification assigned by Supplier (lowest level of packaging which has a package ID code; shall contain like items)
25S	Identification of a party to a transaction as identified in 18V, followed by the supplier assigned serial number
1T	Traceability Number assigned by the supplier to identify/trace a unique group of entities (e.g., lot, batch, heat)
25T	Identification of a party to a transaction as identified in 18V, followed by the supplier assigned traceability number
18V	Vendor Identification (Vendor ID according to ISO/ IEC 15459 is built up from Issuing Agency Code and Company Identification Number assigned by the IAC to the company)

Table 1 Data identifiers to be used

Depending on the level of hierarchy, different data elements are relevant.
On package level, data on the package like packing list number or a tracking number might be applied as well as information on whether the symbol belongs to a sequence of symbols. On the product level proper product/ item identification is essential as well as vendor identification and valid serial numbers or quantities.

Table 2 shows which data fields have to be used per level of the hierarchy.

DI	Multiple hierarchical structured products		
	Hierarchy level		
	Package	Product	Comp
F	M[9]	M	M
4L	N/A	M	O[11]
nJ[5]	M[9]	N/A	N/A
1P[1]	N/A	M	M
25P[1]	N/A	M	M
Q	N/A	M[4,8]	M[4]
13Q[10]	M if >1	N/A	N/A
S[2,3]	N/A	M	M
3S[5]	M[9]	N/A	N/A
25S[2,3]	N/A	M	M
1T[7]	N/A	M[6]	O[4]
25T[7]	N/A	M[6]	O[4]
18V	M[9]	M	M

(M = Mandatory, O = Optional, N/A = not applicable)
Table 2 Mandatory and optional data fields per level

1 25P or 1P must be provided
2 S or 25S must be provided
3 if there is no serial number (possible only if there is no child object), lotNo has to be provided instead if existing
4 if product/ component has serNo -> quantity is optional but always has to be one; if product/ component does not have serNo-> quantity is mandatory
5 1J (nJ in general) could be an alternative to 3S
6 not mandatory if serial number is provided or no traceability code available
7 1T or 25T can be used equally
8 quantity can only be 1 for hierarchical structured products, quantity can be > 1 if the product does not have components listed
9 if there is only one hierarchical product, the package level is optional but should be provided as the reliability of the information provided by the label is only valid until the original package is opened
10 wait for clarification whether a new/ other DI should be used, in the meantime use it (definition pending)
11 Depending on country of delivery and relating customs provisions, CoO may be required by customs authorities also for the components

4.3.2 Requirements on usage of Data Identifiers

4.3.2.1 Product part numbers

- When 25P is available it is not necessary to provide the 1P data element
- When 18V and 1P are available, 25P is optional
- Every product/component has to have a proper part number assigned by the manufacturer

4.3.2.2 Serial and Lot numbers

- When 25S is available it is not necessary to provide the S data element
- When 18V and S are available, 25S is optional
- Components may or may not have serial numbers and/or lot numbers
- Non hierarchical products can have serial and/ or lot numbers
- Assembled products may have a traceability number ("serial numbers"/ lot numbers)
- If the product within the package is an assembled product which is only made up for purchasing but consists of several products/components which also are fully-fledged products themselves, the assembled product must not carry the same serial number as of one the products/component (as the product/component may be replaced)

4.3.2.3 Package Traceability

- 3S and nJ data identifiers can be used complimentary

4.3.2.4 County of Origin (CoO)

- CoO is optional for components as e.g. the CoO for the components of a notebook is not relevant, but the whole notebook
- Depending on country of delivery and relating customs provisions, CoO may be required by customs authorities also for the components and therefore becomes mandatory

4.3.2.5 Quantity

- Quantity is mandatory for non-serialized assembled products/products/components
- (Q is optional for serialized items)
- If a product/component carries a serial number, its quantity has to be one
- If a product/component or product does not carry a serial number, the quantity of the same type of product/component has to be given (e.g. accessories)

4.3.2.6 Traceability number

- A traceability number could be a Serial No, Lot No or Package No.

4.3.2.7 General

- Additional information/Data elements can always be added by the supplier
- By using 18V DI in addition to 25P DI (or other unique identities), manufacturer/supplier relations can be expressed
- Products/components not relevant for tracing or inventory management can be added to the product package without being listed (i.e. non-serialized)
- The validity of the information describing the content of the package is limited to the lifetime of the package. In other words: When the package is opened products in the package or products/components of an assembled product may be changed and the data describing the content becomes false. In this case the label should be destroyed. Only as long as the package is intact (not broken) the information describing the content is reliable. Therefore the "anchor"/"entry object" can only be the package number and not e.g. a serialized product (this is ok for single non-structured products in a package, e.g. a mobile phone)
- It is not recommended to use the hierarchical structured symbol as described by this guideline for marking of products itself if the products/components of the assembled product may be altered during the lifetime of the assembled product, unless there are processes in place which guarantee that the information encoded in the symbol exactly matches the products/components of the assembled product

Figure 6 shows the conceptual illustration of a product package containing an assembled product made up of different components.

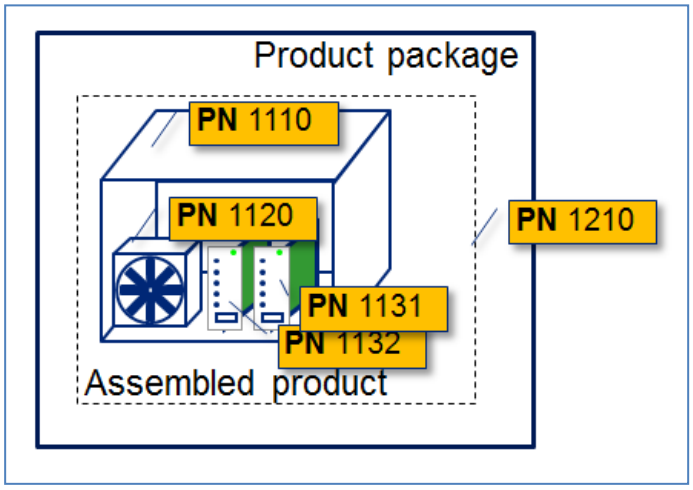


Figure 6 Conceptual illustration of a product package

Figure 7 illustrates the use of ANSI MH 10.8.2 Data Identifiers with an assembled product as introduced in section 3.2.

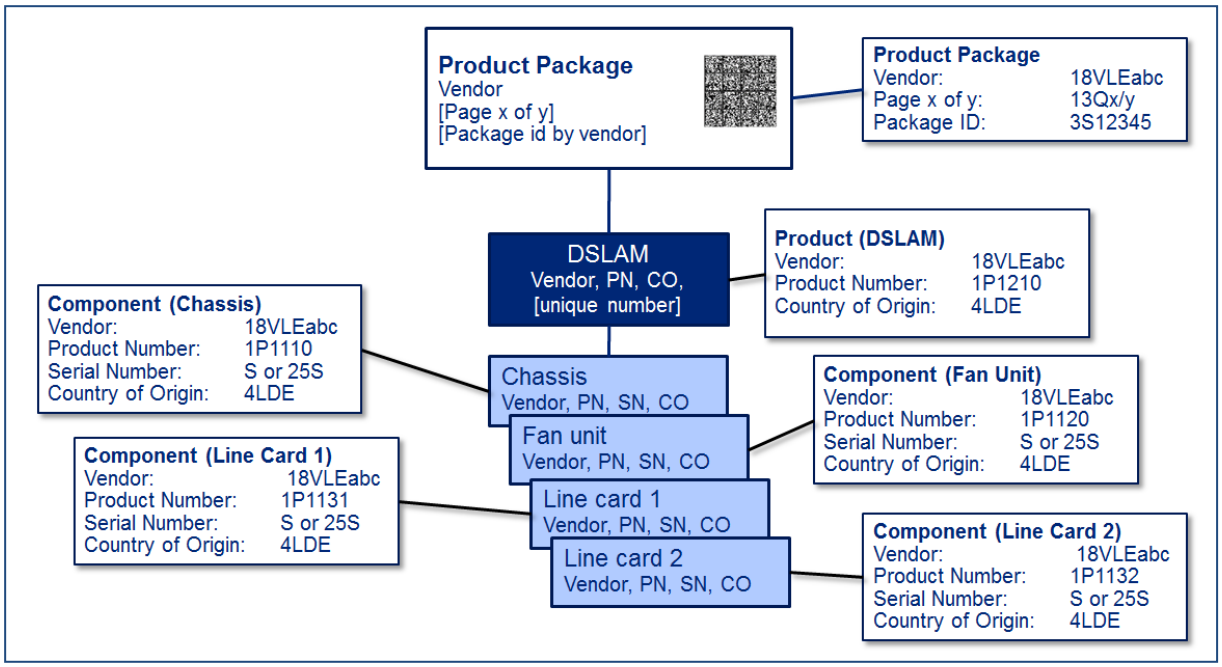


Figure 7 Illustration of ANSI MH 10.8.2 data identifiers

4.4 Data Identifier "F"

This guideline is based on the recommendations of ASC MH 10/SC 8 as described in Section VI of ANSI MH 10.8.2.
The purpose of the DI "F" is to identify dependencies among the content of hierarchically related groups of data segments. The structure of this DI is as follows with all parts required:

Part	String (AN) or Identifier (ID)	Length
DI F	A	1
Hierarchical ID Number	AN	2[1]
Hierarchical Parent ID Number	AN	2[1]
Hierarchical Child Code	ID	1
Hierarchical Level Code	ID	1/2

Note[1] – With the character set of 0-9 and A-Z, a length of 2 characters yields 1,296 permutations

Table 3 ANSI MH10.8.2 Data Identifier "F" Structure

Any "F sequence" contains a parent ID. The link to belonging data will be set by repeating the IDs with the sequences of lower child levels. Additional IDs describe the kind of data such as header information, product data, variables, etc.

Any "F sequence" consists of 7 characters structured as follows:

F	##	##	#	#	Description	Char.
F					Data Identifier "F"	1 a
	##				Hierarchy ID – Number of field	2 an
		##			"Parent ID" – Number of next higher Hierarchy ID	2 an
			#		"Child Code ID" (see table 6 below)	1 n
				#	Hierarchy Level ID (see table 5 below)	1 an

Table 4 DI "F" sequence structure

The Hierarchical levels to be used in the F sequence structure are listed in *Table 6*.

Child Code IDs:

Code	Description
0	No children
1	Has children

Table 5 Child Code IDs

Hierarchical Levels IDs:

Level	HLC	Description
Pack	P	The pack level is used to identify the cartons within which the item is shipped, e.g., label serial numbers. In most cases there will be some sort of packs.
Product (top level Item)	I	Product level - package, stock keeping unit (SKU) identification data (Item).
Component	F	Data related to the manufacturer's component

Table 6 Hierarchy Level Code IDs

The use of hierarchy level "X" representing serial number level is omitted for the first issue of this guideline.

Illustration of the hierarchical logic

The logic of grouping is to repeat the Parent ID of the higher level with the sequence of the connected lower level. The logic contains information like whether or not another level is following and by help of the level code what kind of data it contains.

First level header data, sequence "01": **F01001P**

F	##	##	#	#	Description
F					DI "**F**"
	01				Hierarchy ID of that level: first sequence is "**01**"
		00			**NO Parent ID:** There is no higher level
			1		"Child Code" 0 or 1: **1** indicates that a child is following
				P	Hierarchical level code: "**P**" for pack

Level "02" for item data with product (assembled product): **F02010I**

F	##	##	#	#	Description
F					DI "**F**"
	02				Hierarchy ID of that level for this sequence is: "**02"**
		01			"Parent ID": The ID no. of the next higher hierarchy level is "**01**"
			1		"Child Code" or 1: **1,** YES, there is a child following
				I	Hierarchical level code: "**I**" for products (items)

Level "I" for item data with component 1 Chassis: **F03020C**

F	##	##	#	#	Description
F					DI "**F**"
	03				Hierarchy ID of that level for this sequence is "**03**" (cnd. count)
		02			"Parent ID": The ID no. of the next higher hierarchy level is "**02**"
			0		"Child Code" or 1: **0**, NO there is no more child
				F	Hierarchical level code: "**F**" for component

Level "I" for item data with component 2 Fan unit: **F04020C**

F	##	##	#	#	Description
F					DI "**F**"
	04				Hierarchy ID of that level for this sequence is "**04**" (cnd. count)
		02			"Parent ID": The ID no. of the next higher hierarchy level is "**02**"
			0		"Child Code" or 1: **0**, NO there is no more child
				F	Hierarchical level code: "**F**" for component

Level "I" for item data with component 3 line card 1: **F05020C**

F	##	##	#	#	Description
F					DI "**F**"
	05				Hierarchy ID of that level for this sequence is "**05**" (cnd. count)
		02			"Parent ID": The ID no. of the next higher hierarchy level is "**02**"
			0		"Child Code" or 1: **0**, NO there is no more child
				F	Hierarchical level code: "**F**" for component

Level "I" for item data with component 4 line card 2: **F06020C**

F	##	##	#	#	Description
F					DI "**F**"
	06				Hierarchy ID of that level for this sequence is "**06**" (cnd. count)
		02			"Parent ID": The ID no. of the next higher hierarchy level is "**02**"
			0		"Child Code" or 1: **0**, NO there is no more child
				F	Hierarchical level code: "**F**" for component

Figure 8 shows the usage of DI "F" in a more graphical way:

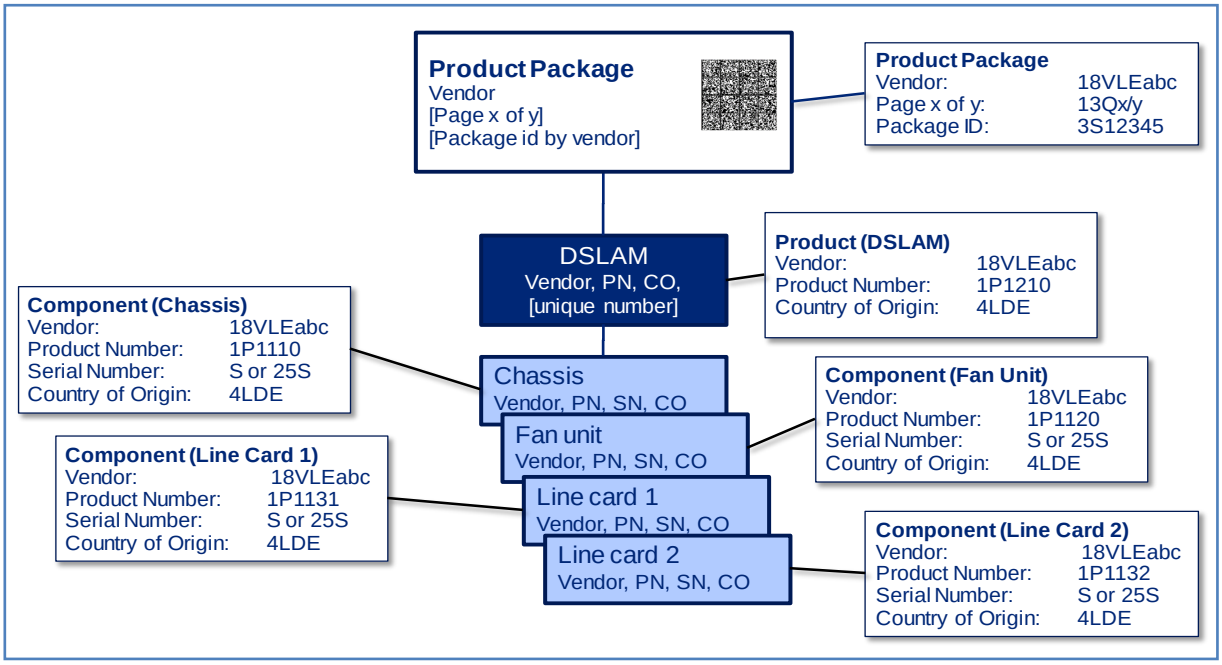


Figure 8 Graphical illustration of the hierarchical logic

4.5 Resulting structure

The elements to code a complete symbol for the marking of hierarchical structured products as described before are:

ISO 15434 Syntax: [ ) > R_S 06 G_S DI DATA G_S DI DATA G_S DI DATA G_S .... G_S DI DATA R_S . E_{OT}

+ Data Elements as listed in chapter 4.3.1
+ Data element "F" structure: F ## ## # #

The full string then looks like the following:

[)>R_S06G_S**F01**001PG_S**18V**LEABCG_S**3S**712245G_S**4L**DEG_S**13Q**1/1G_SF**02**011IG_S**18V**LEABCG_S**1P**1210G_S**Q**1G_S**4L**DEG_S**F03**020FG_S**18V**LEABCG_S**1P**1110G_S**S**0003G_S**4L**DEG_SF**04**020FG_S**18V**LEABCG_S**1P**1120G_S**S**0004G_S**4L**DEG_SF**05**020FG_S**18V**LEABCG_S**1P**1131G_S**S**0005G_S**4L**DEG_SF**06**020FG_S**18V**LEABCG_S**1P**1132 G_S**S**0006G_S**4L**DER_S**E** **O_T**

Figure 9 full symbol data string

5 Application Recommendations

This chapter provides advice on what types of products/applications fit to which scenario and which levels should or should not be included.

5.1 Level of detail to be used with this guideline

The level of detail should be restricted to a certain level to avoid different levels for different customers (e.g. field replaceable units). Products/components must be listed in the package/symbol if they belong to or are above this level.

The lowest level to be used should follow the following recommendations:
The main criteria to determine the lowest level of products/components to be included are

a. whether it is an assembled product or not
b. whether the product/component is under tracing or not (traceability could be defined e.g. by value or importance of the product/component)
c. whether the product/component is field replaceable without any kind of manufacturing procedure or not

Mandatory products/components are:

- serialized products/components, that could (do not require) be replaced in the field without any kind of manufacturing procedure
- the highest level of an assembled product

Optional for the listing in the symbol are e.g.

- sublevels of an assembled product
- products/components which can be purchased on their own/as a separate product/component, but are not serialized
- an installation kit as a whole (containing screws, brackets…), but not the single subcomponents of the installation kit
- a cable kit as a whole (containing different cables), but not the single cables of the cable kit
- important cables/cables of high value (which ought to be serialized)

Notes:

- *The products/components used to form a product are listed in the bill of materials/definition for the assembled product as defined by the manufacturer but could be configured based on available options by the customer*
- *Non-serialized products/components are to be added optionally (choice of supplier)*
- *Only those products/components which have to be tracked should be included to limit the number of products/components to be encoded*
- *In this guideline, a component is regarded as the lowest level that not will be broken down any further*

5.2 Application of Hierarchy Level Codes

It is important to consider that products themselves can contain other products. An item that is contained in an assembled product as a component also could be sold on its own as a product. In the same way, a product which is sold on its own also can be part of an assembled product and therein be looked at as a component. This does not change the

item itself but the hierarchy level code to be used could be different (HLC, also see Table 6). An item might be assigned "F" if seen as component or "I" if seen as a product.
Due to this, it is recommended to not do analysis on the hierarchical level codes "I" and "F" (illustrative example: do not derive number of components by counting all "F" DIs)
The packages on the other hand always receive a hierarchy level code "P".
During application the number of hierarchy levels should be restricted to the necessary minimum to limit complexity.

5.3 Validity of the label

As the label does reflect the original configuration of the product package, it is valid until the original configuration is broken/changed. As soon as the original configuration is broken/changed, it instantly loses validity.

5.4 Label Design and Placement

The 'set label code' - if applicable - should be part of the Product Package Label. On details about the general label design and label location please refer to the EDIFICE Guideline for Product Package Labels, available on the EDIFICE repository.

6 Examples

The following chapters give examples for hierarchical structured products/products assembled from different products/components (scenario 6.1) and multiple products without structured products/components within one package (scenario 6.2). The examples include pictures with detailed product data, the usage of the DI "F", code sequences (symbol data) and illustrate a barcode label.

6.1 Example 1: Scenario "A" - Hierarchical structured products/products assembled from different components

The example shows two DSLAMs of different type in one package including its components.

6.1.1 Package and conceptual representation

Figure 10 illustrates the example product containing two different assembled products, whereas each assembled product is hierarchically structured, and the conceptual representation including the relevant data identifiers and fields.

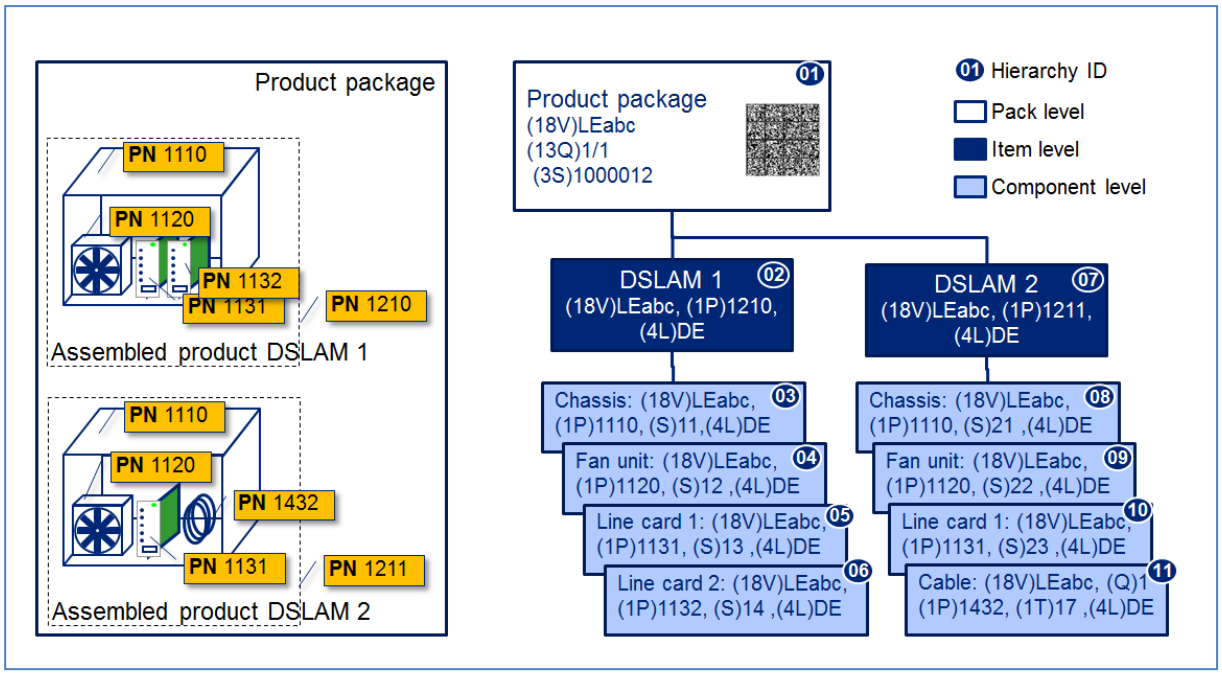


Figure 10 Example 1 - conceptual view

6.1.2 Hierarchy data

The (F) Data Identifiers are shown in the following table:

F	Level ID	Parent ID	Child Code	Hierarchical level	Description
F	**01**	**00**	**1**	**P**	Pack
F	**02**	**01**	**1**	**I**	Item DSLAM 1
F	**03**	**02**	**0**	**F**	Component Chassis Type 1110
F	**04**	**02**	**0**	**F**	Component Fan unit Type 1120
F	**05**	**02**	**0**	**F**	Component Line card 1 Type 1131
F	**06**	**02**	**0**	**F**	Component Line card 1 Type 1132
F	**07**	**01**	**1**	**I**	Item DSLAM 2
F	**08**	**07**	**0**	**F**	Component Chassis Type 1110
F	**09**	**07**	**0**	**F**	Component Fan unit Type 1120
F	**10**	**07**	**0**	**F**	Component Line card 1 Type 1131
F	**11**	**07**	**0**	**F**	Component Cable Type 1432

Table 7 Example 1 - DI F structure

6.1.3 Symbol data

The full string then looks like the following:

[)>RS06GS**F**01001PGS**18V**LEABCGS**13Q**1/1GS**3S**1000012GS**F**02011IGS**18V**LEABCGS**1P**1210GS**4L**DEGS**F**03020FGS**18V**LEABCGS**1P**1110GS**S**11GS**4L**DEGS**F**04020FGS**18V**LEABCGS**1P**1120GS**S**12GS**4L**DEGS**F**05020FGS**18V**LEABCGS**1P**1131GS**S**13GS**4L**DEGS**F**06020FGS**18V**LEABCGS**1P**1132GS**S**14GS**4L**DEGS**F**07011IGS**18V**LEABCGS**1P**1211GS**4L**DEGS**F**08070FGS**18V**LEABCGS**1P**1110GS**S**21GS**4L**DEGS**F**09070FGS**18V**LEABCGS**1P**1120GS**S**22GS**4L**DEGS**F**10070FGS**18V**LEABCGS**1P**1131GS**S**23GS**4L**DEGS**F**11070FGS**18V**LEABCGS**Q**1GS**1P**1432GS**1T**17GS**4L**DERS**E**OT

Figure 11 Example 1 - full symbol data string

6.1.4 Package label

In the following a package label is shown which could be used on the product package. There is a separate area consisting of a package number in plain text and a 2D symbol[2] with the structure of the content in the package.

Optionally the content of the package can be listed as plain text, bar code and/or 2D symbol in the body of the label. To avoid confusion at customs authorities and customers not used to this kind of structures it is strongly recommended NOT to list the products/components of assembled products on the label, i.e. only list the highest level.

[2] Note: the Data Matrix contains illustrative data which does not reflect the complete structure as defined by this guideline

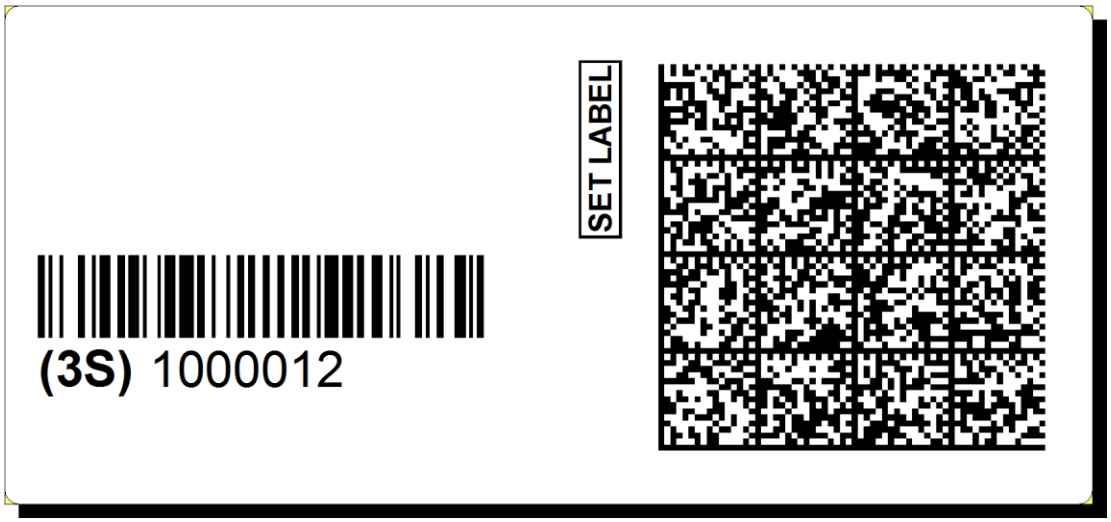


Figure 12 Example 1 - package label for two products in one package

- Level: Packet
  - Location Company<LEABC>, Piece<1>, Number of pieces<1>, Package identification number<1000012>
  - Level: Item
    - Location Company<LEABC>, Article<1210>, Country of Origin<DE>
    - Level: Component
      - Location Company<LEABC>, Article<1110>, Serial number<11>, Country of Origin<DE>
    - Level: Component
      - Location Company<LEABC>, Article<1120>, Serial number<12>, Country of Origin<DE>
    - Level: Component
      - Location Company<LEABC>, Article<1131>, Serial number<13>, Country of Origin<DE>
    - Level: Component
      - Location Company<LEABC>, Article<1132>, Serial number<14>, Country of Origin<DE>
  - Level: Item
    - Location Company<LEABC>, Article<1211>, Country of Origin<DE>
    - Level: Component
      - Location Company<LEABC>, Article<1110>, Serial number<21>, Country of Origin<DE>
    - Level: Component
      - Location Company<LEABC>, Article<1120>, Serial number<22>, Country of Origin<DE>
    - Level: Component
      - Location Company<LEABC>, Article<1131>, Serial number<23>, Country of Origin<DE>
    - Level: Component
      - Location Company<LEABC>, Quantity<1>, Article<1432>, Lot<17>, Country of Origin<DE>

Figure 13 Example 1 - tree representation

Figure 14 Example 1 - package label for one product in one package

6.2 Example 2: Scenario "A" - products made up of multiple hierarchies

The example shows the assembled product "DSLAM" that consists of the different items and components chassis, power supply and line card. The item power supply includes the components fan unit and control unit. So in this example the product is made up of two different hierarchies.

6.2.1 Package and conceptual representation

Figure 15 illustrates the example product for the concept of multiple hierarchies. The conceptual representation of the package includes the relevant data identifiers and fields.

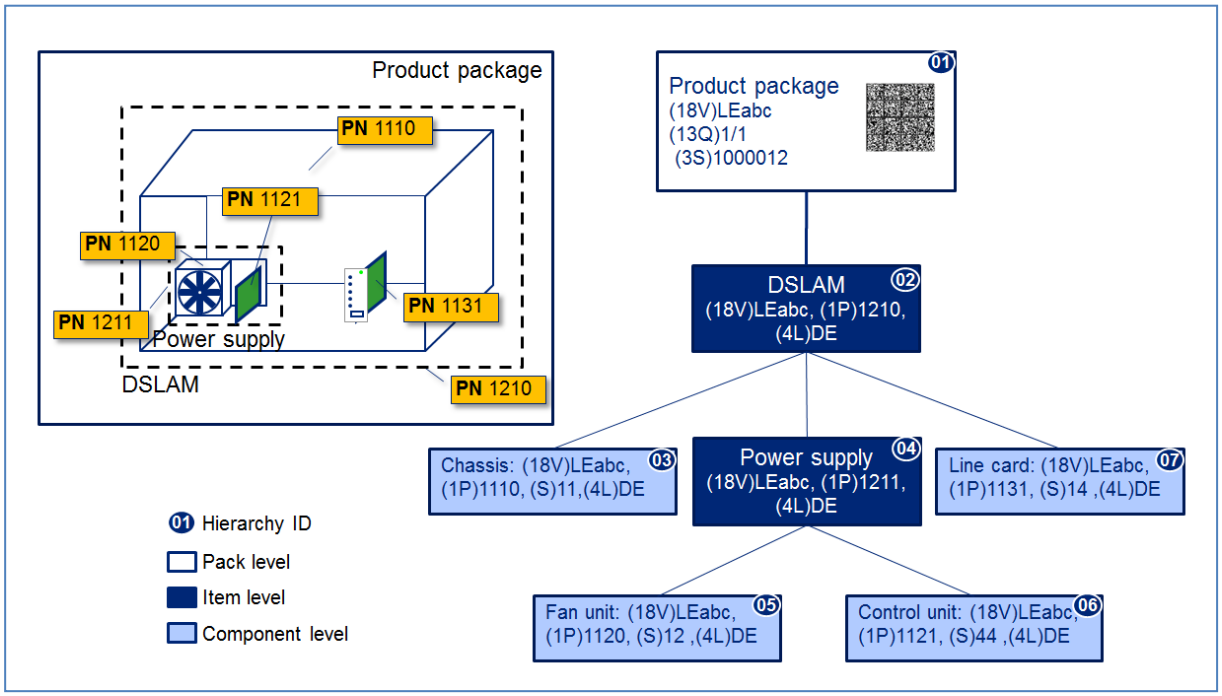


Figure 15 Example 2 - conceptual view

6.2.2 Hierarchy data

The (F) Data Identifiers are shown in the following table:

F	Level ID	Parent ID	Child Code	Hierarchical level	Description
F	**01**	**00**	**1**	**P**	Pack
F	**02**	**01**	**1**	**I**	Item DSLAM
F	**03**	**02**	**0**	**F**	Component Chassis Type 1110
F	**04**	**02**	**1**	**I**	Item Power Supply Type 1211
F	**05**	**04**	**0**	**F**	Component Fan unit Type 1120
F	**06**	**04**	**0**	**F**	Component Control Unit Type 1121
F	**07**	**02**	**0**	**F**	Component Line card Type 1131

Table 8 Example 2 - DI F structure

6.2.3 Symbol data

The full string then looks like the following:

[)>RS06GS**F**01001PGS**18V**LEABCGS**13Q**1/1GS**3S**1000012GS**F**02011IGS**18V**LEABCGS**1P**1210GS**4L**DEGS**F**03020FGS**18V**LEABCGS**1P**1110GS**S**11GS**4L**DEGS**F**04021IGS**18V**LEABCGS**1P**1211GS**4L**DEGS**F**05040FGS**18V**LEABCGS**1P**1120GS**S**12GS**4L**DEGS**F**06040FGS**18V**LEABCGS**1P**1121GS**S**44GS**4L**DEGS**F**07020FGS**18V**LEABCGS**1P**1131GS**S**14GS**4L**DERS**E**OT

Figure 16 Example 2 - full symbol data string

6.2.4 Package label

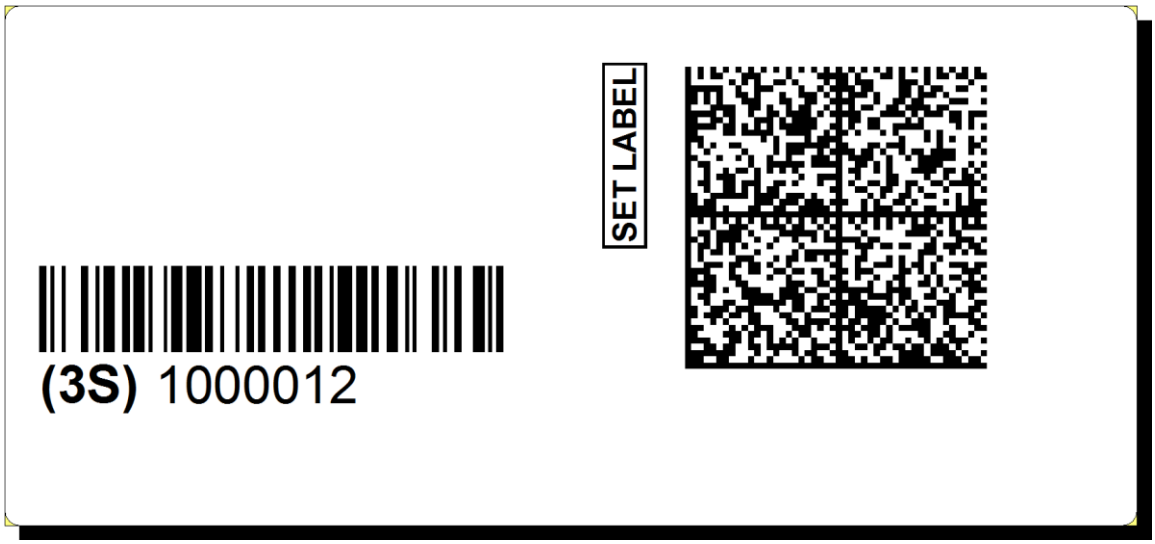


Figure 17 Example 2 - package label for products made up of multiple hierarchies

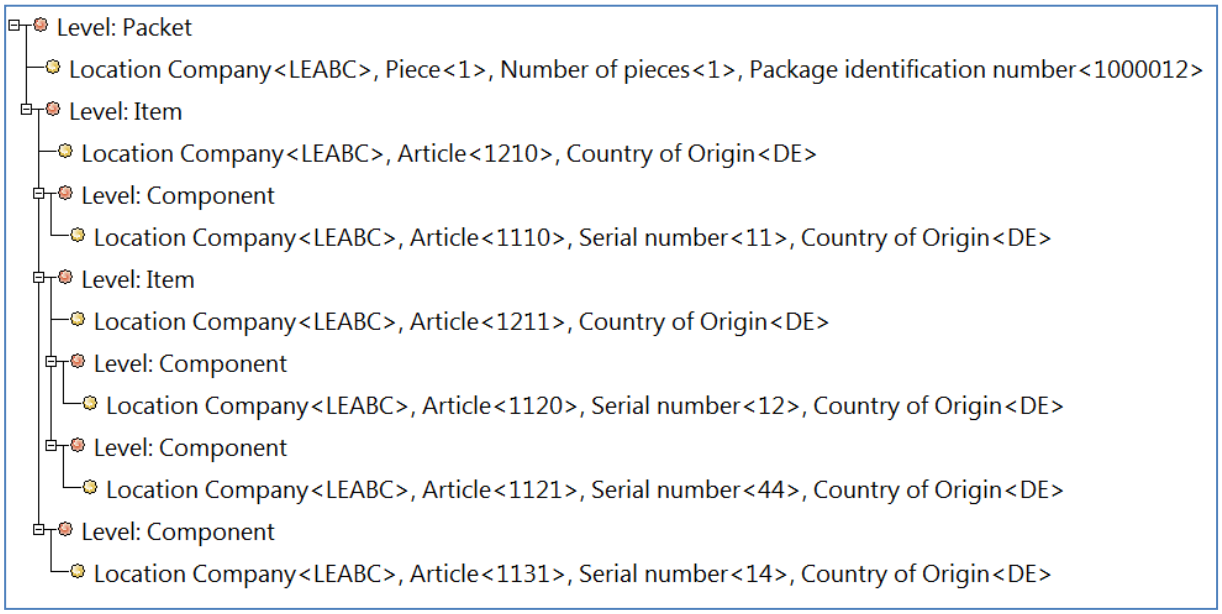


Figure 18 Example 2 - tree representation

6.3 Example 3: Scenario "B" - multiple products of the same type without hierarchy within one package

A package can also contain multiple products that do not have structured products/components listed/visible. A product holding too many components for to be encoded in one Data Matrix can be resolved by the usage of multiple Data Matrix codes. Lot traced products do not require such a structured representation but this kind of structured representation is only adequate for serialized products.
The only mandatory header element in this case is the package number with DI "3S" + "18V".

6.3.1 Package and conceptual representation

Figure 19 illustrates the example product for the concept of multiple products without hierarchy. The conceptual representation of the package includes the relevant data identifiers and fields.

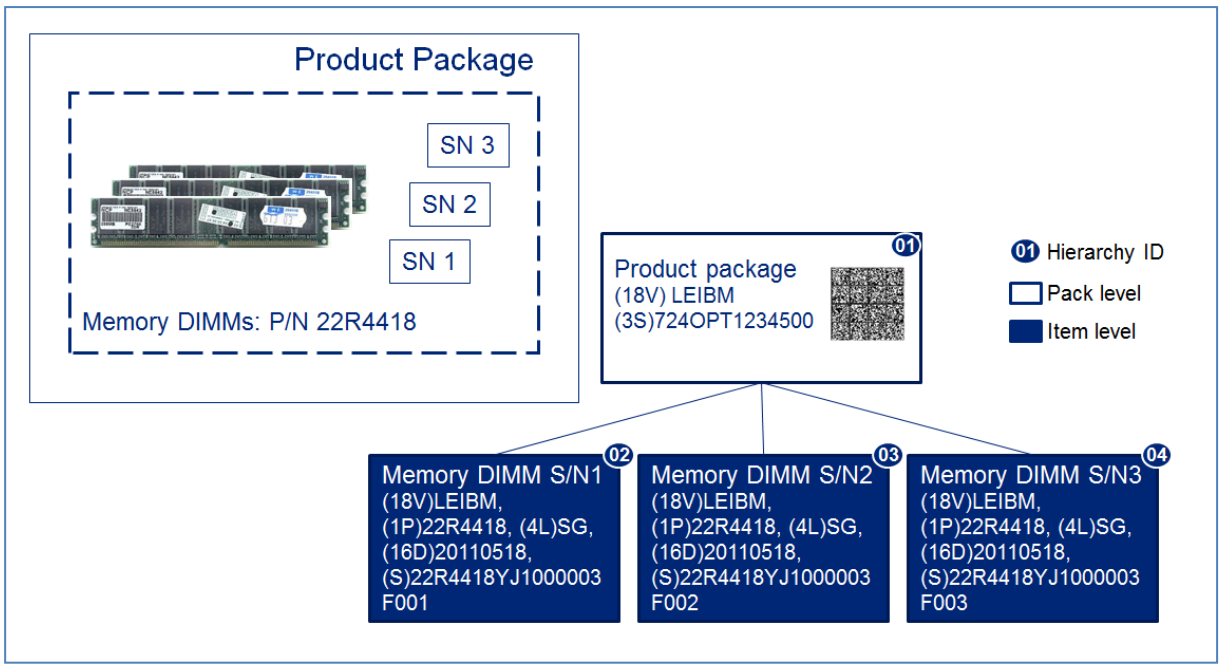


Figure 19 Example 3 - conceptual view

6.3.2 Hierarchy data

F	Level ID	Parent ID	Child Code	Hierarchical level	Description
F	**01**	**00**	**1**	**P**	Pack
F	**02**	**01**	**0**	**I**	Item DRAM 1
F	**03**	**01**	**0**	**I**	Item DRAM 2
F	**04**	**01**	**0**	**I**	Item DRAM 2

Table 9 Example 3 - DI F structure

6.3.3 Symbol data

The full string then looks like the following:

[)>[R]S06[G]S**F**01001P[G]S**18V**LEIBM[G]S**3S**724OPT1234500[G]S**F**02010I[G]S**18V**LEIBM[G]S**1P**22R44
18[G]S**S**22R4418YJ100003F001[G]S**4L**SG[G]S**16D**20110518[G]S**F**03010I[G]S**18V**LEIBM[G]S**1P**22R44
18[G]S**S**22R4418YJ100003F002[G]S**4L**SG[G]S**16D**20110518[G]S**F**04010I[G]S**18V**LEIBM[G]S**1P**22R44
18[G]S**S**22R4418YJ100003F003[G]S**4L**SG[G]S**16D**20110518 [R]S**E**OT

Figure 20 Example 3 - full symbol data string

6.3.4 Package label

The 2D symbol contains the header-data of the package ("3S", "18V", and eventually other optional data elements) and the content data of the package.
Optionally all the content in the package can be listed as plain text, bar code and/or 2D symbol in the body of the label.

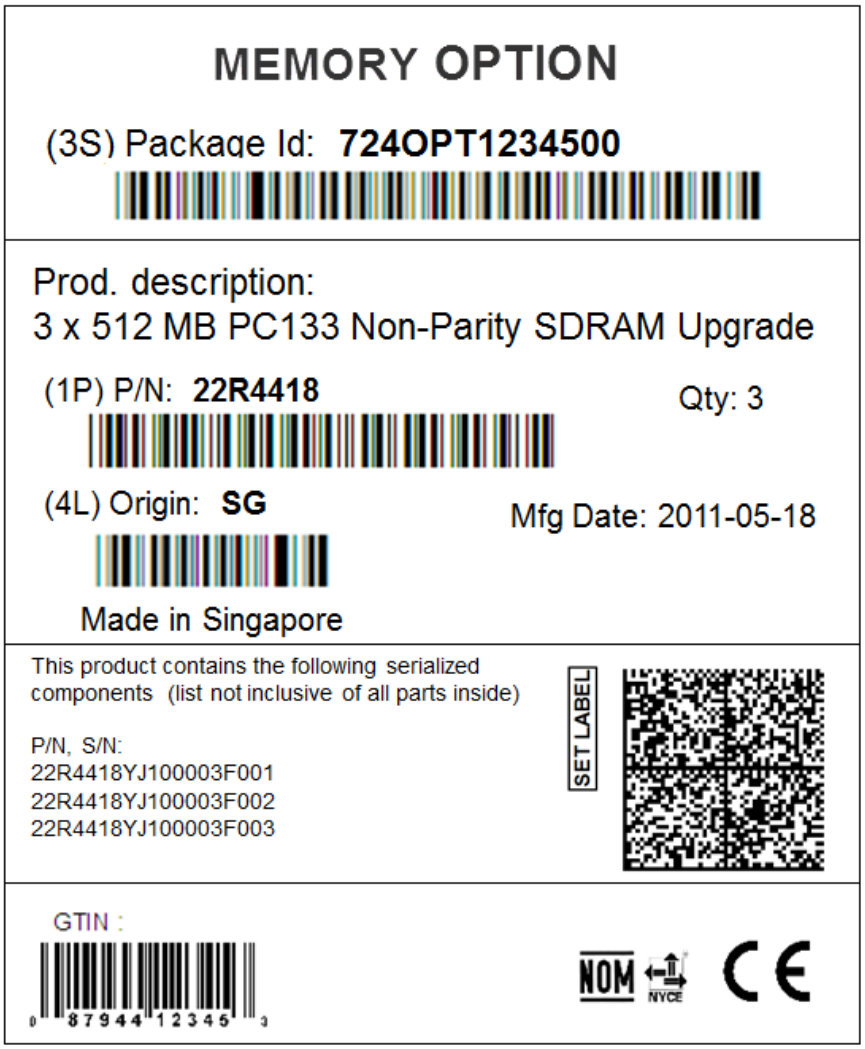


Figure 21 Example 3 - illustration of a possible package label

Note: The presented label in figure 21 is illustrative only. In the long run only the part containing the set label symbol will be applicable. The Barcodes representing single products are presented only as a migration path. Their information is redundant to the set label symbol but allows companies not yet able to scan the set label to capture the component data.

Figure 22 Example 3 - tree representation

6.4 Example 4: Scenario "B" – multiple products of different types without hierarchy which are packaged in one package

Example 4 describes a product package containing a mobile phone and its accessories. The package is primarily connected to the included phone and its serial number - as that is the serial number that is of interest when to register the purchase and setting warranty for the "phone".

6.4.1 Package and conceptual representation

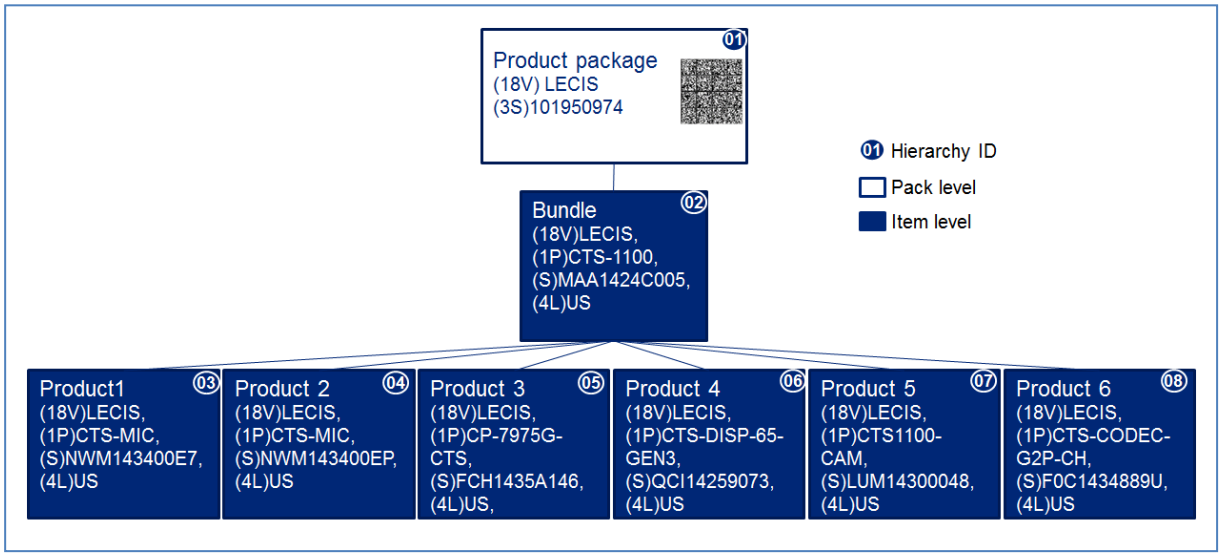


Figure 23 Example 4 - package illustration

So in this case a content specification would look like that the package F includes Product A to E of which A, C and E are serialized and of which serial number for Product A also is "duplicated" on the package.

6.4.2 Hierarchy data

F	Level ID	Parent ID	Child Code	Hierarchical level	Description
F	**01**	**00**	**1**	**P**	Pack
F	**02**	**01**	**0**	**I**	Bundle Item
F	**03**	**02**	**0**	**I**	Item 1
F	**04**	**02**	**0**	**I**	Item 2
F	**05**	**02**	**0**	**I**	Item 3
F	**06**	**02**	**0**	**I**	Item 4
F	**07**	**02**	**0**	**I**	Item 5
F	**04**	**02**	**0**	**I**	Item 6

Table 10 Example 4 - DI F structure

6.4.3 Symbol data

The full string then looks like the following:

[)>$^{R}_{S}$ 06$^{G}_{S}$**F**01001P$^{G}_{S}$**18V**LECIS$^{G}_{S}$**3S**1019509 74$^{G}_{S}$**F**02010I$^{G}_{S}$**18V**LECIS$^{G}_{S}$**1P**CTS-1100$^{G}_{S}$**S**MAA1424C005$^{G}_{S}$**4L**US$^{G}_{S}$F03020I$^{G}_{S}$**18V**LECIS$^{G}_{S}$**1P**CTS-MIC$^{G}_{S}$**S**NWM143400E7$^{G}_{S}$**4L**US$^{G}_{S}$F04020I$^{G}_{S}$**18V**LECIS$^{G}_{S}$**1P**CTS-MIC$^{G}_{S}$**S**NWM143400EP$^{G}_{S}$**4L**US$^{G}_{S}$F05020I$^{G}_{S}$**18V**LECIS$^{G}_{S}$**1P**CP-7975G-CTS$^{G}_{S}$**S**FCH1435A146$^{G}_{S}$**4L**US$^{G}_{S}$23SE80462EB9C00$^{G}_{S}$**F**06020I$^{G}_{S}$**18V**LECIS$^{G}_{S}$**1P**CTS-DISP-65-GEN3$^{G}_{S}$**S**QCI14259073$^{G}_{S}$**4L**US$^{G}_{S}$F07020I$^{G}_{S}$**18V**LECIS$^{G}_{S}$**1P**CTS1100-CAM$^{G}_{S}$**S**LUM14300048$^{G}_{S}$**4L**US$^{G}_{S}$F08020I$^{G}_{S}$**18V**LECIS$^{G}_{S}$**1P**CTS-CODEC-G2P-CH$^{G}_{S}$**S**FOC1434889U $^{G}_{S}$**4L**US$^{R}_{S}$ $^{E}_{OT}$

Figure 24 Example 4 - full symbol data string

6.4.4 Package label

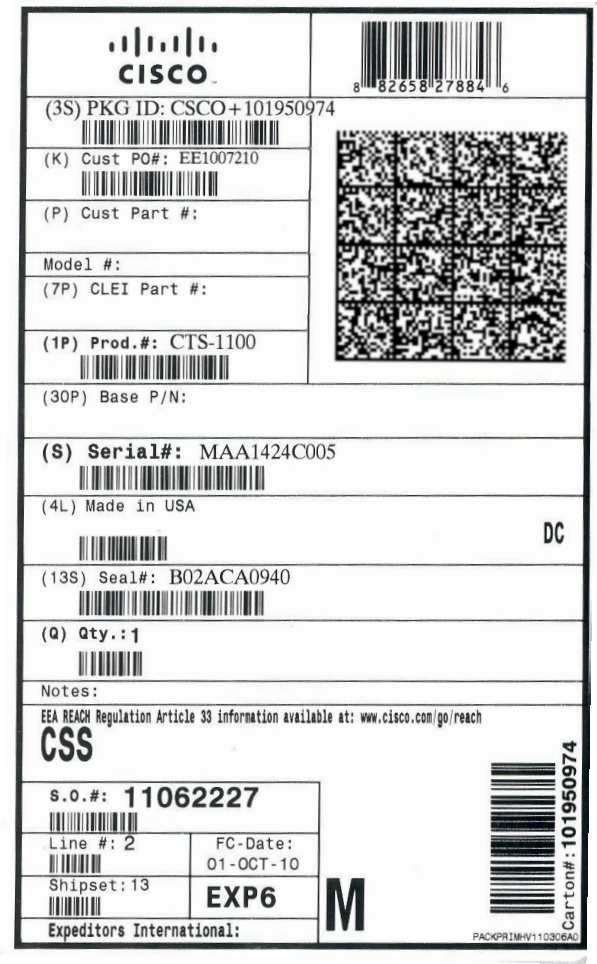


Figure 25 Example 4 - bundle label

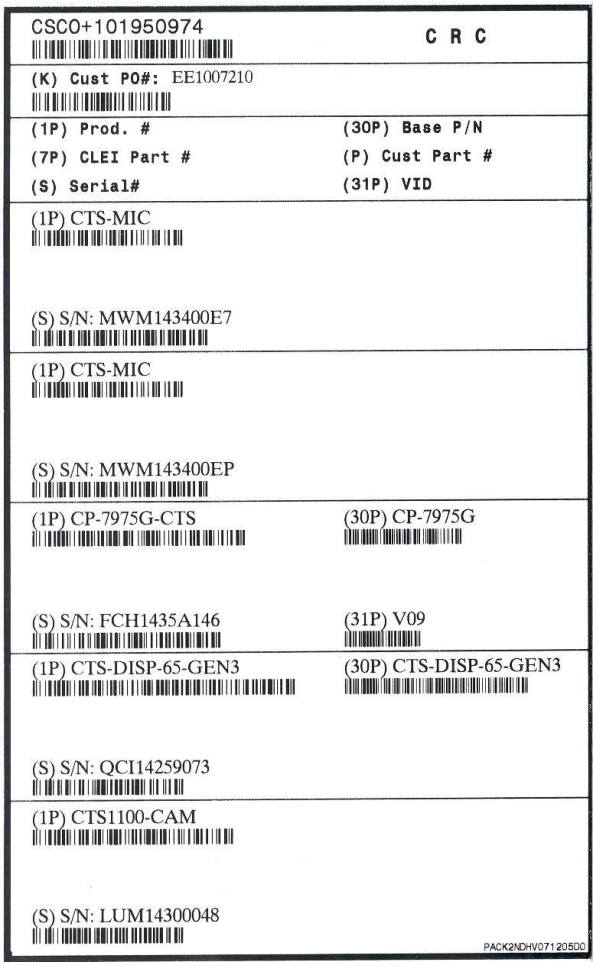


Figure 26 Example 4 - content label 1

Figure 27 Example 4 - content label 2

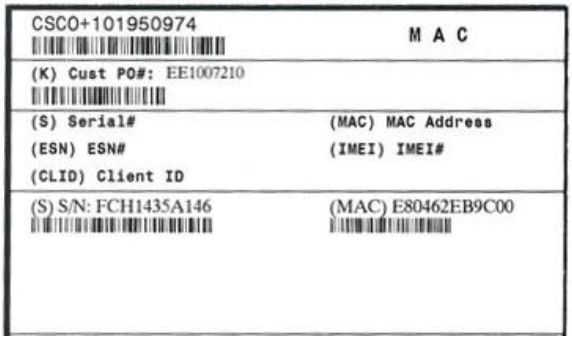


Figure 28 Example 4 - MAC-address

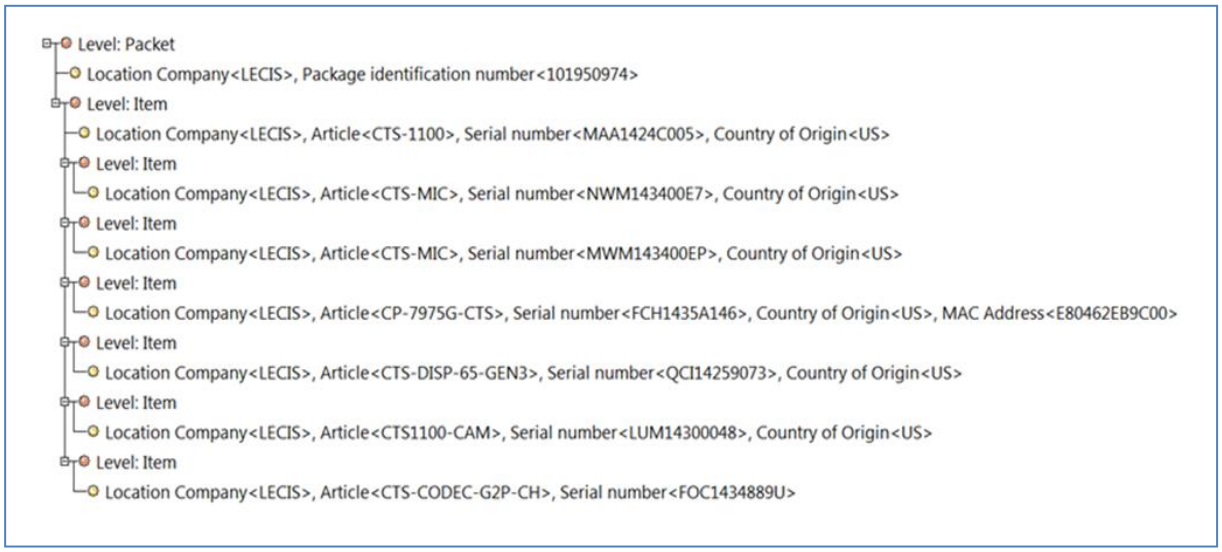


Figure 29 Example 4 - tree representation

7 Appendix A - recommendations on data carriers

7.1 Data Matrix

7.1.1 Symbol sizes and print resolution

Symbol sizes are determined by the dimension of a single module of a matrix (X dimension) and by the data volume. The permitted matrix sizes (x modules by y modules) are specified with ISO/IEC 16022. It is recommended not to exceed a symbol matrix of 96 x 96 modules due to the performance of scanners available on the date of publication of this document. A minimum nominal X dimension of 0.25mm (10mil) is recommended.

7.1.2 Symbol quality

Symbol quality for labels shall be measured according to ISO/IEC 15415. The Grade "C" ("2") is recommended as the overall quality grade with the label applied to the package.

7.1.3 Use of "Macro 06"

To reduce the symbol size when using DIs, the "Macro 06" is defined for use with Data Matrix. It allows reducing the syntax header (start sequence [)>^{R_S}06^{G_S} and trailer data ^{R_S} $^{E}O_T$) to one Data matrix codeword (code 237) defined in ISO/IEC 16022. When used the reader will extrapolate the normal sequence of characters from the single codeword already on the symbol/ physical layer and thereby the macro is transparent to the user. The Macro 06 codeword is supported by printing software and scanners compliant with the ISO/IEC 16022 standard.