The Global Language of Business

GS1 Digital Link Standard: URI Syntax

Enabling consistent representation of GS1 identification keys within web addresses to link to online information and services

Release 1.3~~1.2~~, ~~Ratified, Feb 2022~~Community Review Draft, Aug 2022

Document Summary

Document Item	Current Value
Document Name	GS1 Digital Link Standard: URI Syntax
Document Date	
Document Version	1.3~~1.2~~
Document Issue	0~~1~~
Document Status	
Document Description	Enabling consistent representation of GS1 identification keys within web addresses to link to online information and services

Contributors

Name	Company
Kishore Karuppan (Chair)	Procter & Gamble Co.
Michel Ottiker (Chair)	GS1 Switzerland
Dominique Guinard (Chair)	EVRYTHNG
Stefan Artlich	Bayer AG - Division Pharma
Adam Björnstjerna	HKScan Sweden AB
Wes Bloemker	Arthrex Inc.
Stephen Brown	Mead Westvaco
Greg Buckley	PepsiCo, Inc.
Jeanne Duckett	Avery Dennison RFID
Vera Feuerstein	Nestlé
Tarik Gemei	Amgen Inc.
Viktoria Grahnen	AbbVie
Zoltan Homan	Cook Medical Inc.
Sinead Kennedy	Cook Medical Australia
Mike Kuhno	Avery Dennison RFID
Nicolas Lecocq	L'Oreal
Phill Marley	AstraZeneca Pharmaceuticals
Fritz Mbumb-Kumb	EM Microelectronic
Yi Meng	Qingdao Haier Washing Machine Co. Ltd.
Paul Muller	EM Microelectronic
Martin Neselius	BillerudKorsnäs (Venture)
Tatjana Pathare	F. Hoffmann-La Roche Ltd.
Mandeep Sodhi	Nestlé
Jim Springer	EM Microelectronic
John Terwilliger	Abbott
Gina Tomassi	PepsiCo, Inc.
Jeroen van Rosmalen	Amgen Inc.

Name	Company
Julie Vargas	Avery Dennison RFID
Sylvie Vilcoq	DANONE PRODUITS FRAIS FRANCE
Evan Bacchus	Costco Wholesale
Fabien Calleia	SIZEASE
Sven Dienelt	Hermann Hagemeyer GmbH & Co. KG
Marcel Ducceschi	Migros-Genossenschafts-Bund
Jonas Elander	Axfood Sverige AB
Max Engström	H&M
Plamen Iliev	DECATHLON
Jerome Lemay	DECATHLON
Dibyajeeban Mishra	DECATHLON
Sylvia Rubio Alegren	ICA Sverige AB
Hans Peter Scheidt	C & A SCS
Martijn Veerman	Customer Value
Joachim Wilkens	C & A SCS
Jeff Denton	AmerisourceBergen Corporation
Vladimir Dzalbo	Smartrac Technology Germany GmbH
Richard Fisher	DoD Logistics AIT Standards Office
Hajo Reissmann	Universitaetsklinikum Schleswig-Holstein
Hirokazu Nagai	Japan Pallet Rental Corporation
Thomas Burke	Institute of Food Technologists
Albert Arbones	GS1 Spain
Karen Arkesteyn	GS1 Belgium & Luxembourg
Andrea Arozamena	GS1 Mexico
Koji Asano	GS1 Japan
Andrea Ausili	GS1 Italy
Mahdi Barati	GS1 Iran
Xavier Barras	GS1 France
Jonas Batt	GS1 Switzerland
Arnaud Bonnefoy	GS1 France
Jonas Buskenfried	GS1 Sweden
Emanuela Casalini	GS1 Italy
Madalina Cernat	GS1 Romania
Anthony Chan	GS1 Hong Kong, China
Shawn Chen	GS1 Thailand
Luiz Costa	GS1 Brasil
Benjamin Couty	GS1 France
Amanda Creane	GS1 Ireland
Tim Daly	GS1 Ireland
Owen Dance	GS1 New Zealand
Michael Davis	GS1 Australia

Name	Company
Kevin Dean	GS1 Canada
Huipeng Deng	GS1 China
Sean Dennison	GS1 Ireland
Peta Ding	GS1 UK
Deniss Dobrovolskis	GS1 Sweden
Nipun Dogra	GS1 India
Xiaowen Dong	GS1 China
Gianluca Fazio	GS1 Argentina
Guilherme França	GS1 Brasil
Michele Francis Padayachee	GS1 South Africa
Jesper Kervin Franke	GS1 Denmark
Jean-Christophe Gilbert	GS1 France
Vanessa Giulieri	GS1 Italy
Alvin Goh	GS1 Singapore
Nicole Golestani	GS1 Canada
Heinz Graf	GS1 Switzerland
Magali Granger	GS1 France
Marija Groznik Stankovic	GS1 Slovenia
János Gyuris	GS1 Hungary
Rami Habbal	GS1 UAE
Jason Hale	GS1 UK
Gary Hartley	GS1 New Zealand
Sandra Hohenecker	GS1 Germany
Hideki Ichihara	GS1 Japan
Yoshihiko Iwasaki	GS1 Japan
Yohan Jeon	GS1 Korea
Fiona (Zhitao) Jia	GS1 China
Sang Ik Jung	GS1 Korea
Iliada Karali	GS1 Association Greece
Kimmo Keravuori	GS1 Finland
Kazuna Kimura	GS1 Japan
Dora Kit	GS1 Hong Kong, China
Sabine Klaeser	GS1 Germany
Alexey Krotkov	GS1 Russia
Chris Lai	GS1 Hong Kong, China
Ildikó Lieber	GS1 Hungary
Xiaoyan Liu	GS1 China
Marisa Lu	GS1 Chinese Taipei
Ilka Machemer	GS1 Germany
Noriyuki Mama	GS1 Japan
Roberto Matsubayashi	GS1 Brasil

Name	Company
Riad Mechtari	GS1 Algeria
Terje Menkerud	GS1 Norway
Jan Merckx	GS1 Netherlands
Ephraim Mokheseng	GS1 South Africa
Adrien Molines	GS1 France
Naoko Mori	GS1 Japan
Daniel Mueller-Sauter	GS1 Switzerland
Prince Namane	GS1 South Africa
Jorge Andrés Nava Alanís	GS1 Mexico
Zubair Nazir	GS1 Canada
Daisuke Negishi	GS1 Japan
Alice Nguyen	GS1 Vietnam
Maciej Niemir	GS1 Poland
Staffan Olsson	GS1 Sweden
Manos Papadakis	GS1 Association Greece
Sebastián Perazzo	GS1 Argentina
Thiago Perez Rojas	GS1 Argentina
James Perng	GS1 Chinese Taipei
Bijoy Peter	GS1 India
Sarina Pielaat	GS1 Netherlands
Aruna Ravikumar	GS1 Australia
Paul Reid	GS1 UK
Zbigniew Rusinek	GS1 Poland
Nick Rusman	GS1 Netherlands
Sunny Sanam	GS1 Australia
Roxana Saravia Bulmini	GS1 Argentina
Yuki Sato	GS1 Japan
Sue Schmid	GS1 Australia
Eugen Sehorz	GS1 Austria
Pooja Sengupta	GS1 Australia
Xiaojing Shao	GS1 China
Yuko Shimizu	GS1 Japan
Marcel Sieira	GS1 Australia
Cesar Silvestre	GS1 Mexico
Olga Soboleva	GS1 Russia
Roko Staničić	GS1 Slovenia
Andrew Steele	GS1 Australia
Sylvia Stein	GS1 Netherlands
Jo Anna Stewart	GS1 US
Ralph Troeger	GS1 Germany
Frits van den Bos	GS1 Netherlands

Name	Company
Ricardo Verza Amaral Melo	GS1 Brasil
Linda Vezzani	GS1 Italy
Rocio Vizcarra	GS1 Argentina
Amber Walls	GS1 US
Yi Wang	GS1 China
Achim Wetter	GS1 Germany
Stephan Wijnker	GS1 Australia
Dirk Willekens	GS1 Belgium & Luxembourg
Connie Wong	GS1 Canada
Ruoyun Yan	GS1 China
Shawn Zhang	GS1 China
Victor Zhang	GS1 China
Dieter Beitz	CSB System AG
Marc Blanchet	Viagenie
Shreenidhi Bharadwaj	Syndigo
Scott Brown	1WorldSync, Inc.
Shawn Cady	Syndigo
Ed Collins	Brandbank
J.D. Kern	Syndigo
Sprague Ackley	Digimarc
Adnan Alattar	Digimarc
Philip Allgaier	bpcompass GmbH
Attilio Bellman	Antares Vision
Karim Ben Dakhlia	Dentsu Aegis Network
Jayson Berryhill	Envisible LLC
Dalibor Biscevic	Business Technologies Ltd
Megan Brewster	Impinj, Inc
Menno Bruil	H2Compute
Randy Burd	Kwikee, A Syndigo Company
Steffen Butschbacher	bpcompass GmbH
Tony Ceder	Charmingtrim
Robert Celeste	Center for Supply Chain Studies
Patrick Chanez	INEXTO SA
Grant Courtney	Be4ward ltd
Henk Dannenberg	NXP Semiconductors
Dilip Daswani	Qliktag Software (formally Zeebric LLC)
Cory Davis	Digimarc
Christophe Devins	Adents
Roland Donzelle	SQUARE / TINTAMAR
Chuck Evanhoe	Evanhoe & Associates, Inc.
Susan Flake	Zebra Technologies Corporation

Name	Company
Tomaz Frelih	Četrta pot,d.o.o.,Kranj
Mathieu Gallant	Optel Group
Ivan Gonzalez	recycl3R
Richard Graves	Phy
Danny Haak	Nedap
Steve Halliday	RAIN RFID Alliance
Mark Harrison	Milecastle Media Limited
Philip Heggelund	DuckScape Inc
John Herzig	Barcode Graphics Inc Canada
Bernie Hogan	Independent Consultant - Bernie Hogan
Dan James	Digimarc
Sandun Jayawardena	H2Compute
Margo Johnson	Transmute
Paul Kanwar	ScanTrust
Thomas Kühne	Goodstag GmbH
Sean Lockhead	Lockhead Consulting Group LLC
Andrew Love	Be4ward ltd
André Machado	TrustaTAG
Lee Metters	Domino Printing Sciences PLC
Joel Meyer	Digimarc
Mario Mira	Dentsu Aegis Network
Attila Sándor Nagy	infiCom.EU Co. Ltd.
Ilteris Oney	ecomes
Mitun Pandey	Goodstag GmbH
Tiphaine Paulhiac	Ambrosus Technologies
Fernando Pereira	Saphety Level SA
Justin Picard	ScanTrust
Scott Pugh	Jennason LLC
Tony Rodriguez	Digimarc
Octavio Rodriguez	Systech International
Zbigniew Sagan	Advanced Track and Trace
Joannie Sauvageau	Optel Group
Kim Simonalle	Qliktag Software (formally Zeebric LLC)
Laurent Tonnelier	mobiLead
Andrew Verb	Bar Code Graphics, Inc.
Elizabeth Waldorf	TraceLink
Alex Winiarski	Winiarski Group
George Wright IV	Product Identification & Processing Systems
Shi Yu	Beijing REN JU ZHI HUI Technology Co. Ltd.
Pete Alvarez	GS1 Global Office
Phil Archer	GS1 Global Office

Name	Company
Lena Coulibaly	GS1 Global Office
Peta Ding	GS1 Global Office
Nadi (Scott) Gray	GS1 Global Office
Steven Keddie	GS1 Global Office
Neil Piper	GS1 Global Office
Craig Alan Repec	GS1 Global Office
Greg Rowe	GS1 Global Office
John Ryu	GS1 Global Office

Log of Changes

Release	Date of Change	Changed By	Summary of Change
1.0	Aug 2019	Mark Harrison, Phil Archer, Dominique Guinard, Marie Petre & Greg Rowe	Initial release developed on WR 17-000343. Originally published under the title **GS1 Web URI Structure Standard**
1.1	Feb 2020	Mark Harrison, Phil Archer Greg Rowe	Updates based upon WR 18-231 which can be found in section 10
1.2	Jan 2021	Mark Harrison, Phil Archer, Dominique Guinard, Steven Keddie & Greg Rowe	Updates based upon WR 20-127 which can be found in section 8
1.2.1	Feb 2022	Mark Harrison	WR 21-429 bundles WR 21-185 /420/ 421/422. Reference change log *8.1*
1.3.0	Oct 2022 ??	Phil Archer & Peta Ding	WR 22-255 (reinstating AI 415) and WR 22-247 (removal of deprecated convenience alphas)

Disclaimer

GS1®, under its IP Policy, seeks to avoid uncertainty regarding intellectual property claims by requiring the participants in the Work Group that developed this **GS1 Digital Link Standard: URI Syntax** to agree to grant to GS1 members a royalty-free licence or a RAND licence to Necessary Claims, as that term is defined in the GS1 IP Policy. Furthermore, attention is drawn to the possibility that an implementation of one or more features of this Specification may be the subject of a patent or other intellectual property right that does not involve a Necessary Claim. Any such patent or other intellectual property right is not subject to the licencing obligations of GS1. Moreover, the agreement to grant licences provided under the GS1 IP Policy does not include IP rights and any claims of third parties who were not participants in the Work Group.

Accordingly, GS1 recommends that any organisation developing an implementation designed to be in conformance with this Specification should determine whether there are any patents that may encompass a specific implementation that the organisation is developing in compliance with the Specification and whether a licence under a patent or other intellectual property right is needed. Such a determination of a need for licencing should be made in view of the details of the specific system designed by the organisation in consultation with their own patent counsel.

THIS DOCUMENT IS PROVIDED "AS IS" WITH NO WARRANTIES WHATSOEVER, INCLUDING ANY WARRANTY OF MERCHANTABILITY, NONINFRINGEMENT, FITNESS FOR PARTICULAR PURPOSE, OR ANY WARRANTY OTHER WISE ARISING OUT OF THIS SPECIFICATION. GS1 disclaims all liability for any damages arising from use or misuse of this document, whether special, indirect, consequential, or compensatory damages, and including liability for infringement of any intellectual property rights, relating to use of information in or reliance upon this document.

GS1 retains the right to make changes to this document at any time, without notice. GS1 makes no warranty for the use of this document and assumes no responsibility for any errors which may appear in the document, nor does it make a commitment to update the information contained herein.

GS1 and the GS1 logo are registered trademarks of GS1 AISBL.

Table of Contents

91
92

1 Introduction

This section and its subsections are informative

GS1 defines a wide range of identifiers that underpin the supply chain and retail industry across the world. This document assumes the reader is familiar with these and the concept of GS1 Application Identifiers. If not, please see information on [GS1 Identification Keys] and the [GENSPECS] for further background.

This work has been motivated by a number of trends. For example: the desire among retailers to move to 2D barcodes that can carry more information than just the GTIN; the problems of multiple barcodes causing scanning errors through conflicts which suggests a need for a single but multipurpose barcode; the growing expectation among consumers that more information is available online about the products they're considering buying; the brand owner concept of the pack as a media channel linking to multimedia experiences, and more.

As a result of this standard, it is possible to represent GS1 identification keys consistently within Web addresses as well as within barcodes containing Web addresses, such that a single identification approach can support both product identification for supply chain applications *and* a link to online material for consumer and business partner interactions. It's this dual functionality and enormous flexibility that is currently not possible when, for example, Brand Owners embed an unstructured Web page address in a QR Code®[1].

The scope of the work accommodates all Class 1 and Class 2 GS1 Keys and Key qualifiers (e.g., serial number, batch number, consumer product variant) and other relevant attributes as the same technologies are equally applicable to SSCCs, GLNs, GIAIs, GRAIs, GSRNs etc. While the syntax can support Class 2 Keys, it is up to the Class 2 Issuing Agencies to determine whether it's fit for their use. For Class 3 GS1 Keys, GS1 welcomes bilateral discussions with Issuing Agencies to see where alignment is possible.

This GS1 standard references a number of third-party standards from the Internet Engineering Task Force (IETF) and the World Wide Web Consortium (W3C).

1.1 How the GS1 Digital Link standard documents fit together

Rather than one very long document containing every detail, as of version 1.2, the GS1 Digital Link standard comprises 4 discrete documents:

URI syntax (this document)

This document provides some of the background to the design of GS1 Digital Link, highlighting existing techniques and practices that underpin the World Wide Web, and applying those to the GS1 system. The normative portions set out the detailed syntax of Web addresses (HTTP URIs) that encode GS1 identifiers with exactly the same precision and expressivity as the AI-based element syntax used across the GS1 system, notably in the GS1 General Specifications. The GS1 Digital Link URI syntax distinguishes between primary keys, such as GTIN and GLN, key qualifiers, such as batch/lot and GLN extension, and attributes such as expiry date and ship-to address. The GS1 Digital Link URI syntax is the foundation on which all other aspects of the standard are built.

Compression

A GS1 Digital Link URI that contains a set of identifiers and attributes may exceed the capacity of some data carriers. This document defines a compression/decompression algorithm that minimises the length of those Web URIs while retaining two critical features: 1) that the compressed form is still a URL on the same domain as the uncompressed form, that is, there is no change in ownership of the URL; 2) that it can be decompressed and the GS1 keys extracted *without* an online lookup.

Resolution

A GS1 Digital Link URI is a particular form of URL and *can* be used in exactly the same way as any other URL (this is an important design feature). However, it can also be the gateway to multiple sources of information, both human and machine-readable. This document defines how the keys in

[1] Unless otherwise specified, the term 'QR Code®' refers to the widely used *ISO/IEC 18004 QR Code*®, excluding the GS1 QR Code that recognises the FNC1 character. 'QR Code' is a registered trademark of Denso Wave, a subsidiary of Denso Corporation. Both the *ISO/IEC 18004 QR Code*® and GS1 QR Code follow the encoding scheme described in ISO/IEC 18004 Information technology — Automatic identification and data capture techniques — QR Code bar code symbology specification, 3rd edition 2015-02-01.

a GS1 Digital Link URI can be 'resolved' to those information sources in such a way that information systems and apps can discover them automatically. Resolvers are what makes the standard operational for the GS1 community and the industries served.

Semantics

Devices like scanners and point of sale terminals, PIM systems, product catalogues and more that are designed specifically to work with GS1 identifiers and data carriers, are all programmed to function within that particular framework. GS1 Digital Link puts things like GTINs, SSCCs and GRAIs onto the Web alongside countless other identifiers and ways of working. This document expresses the meaning behind the GS1 Digital Link standard in a way that the Web at large can understand and process. It makes use of, and extends, the GS1 Web Vocabulary.

1.2 Typographical conventions used in this document

This document includes a lot of examples of GS1 Digital Link URIs such as:

```
https://example.org/414/{gln}/254/{glnExtension}
```

```
https://example.org/01/{gtin}{?exp}
```

The use of the monospace font indicates that the text has meaning for computers. Further, these examples follow the convention used in [RFC 6570]. The places where the values of variables should be inserted are written in braces, so, for example, {gtin} means "insert GTIN here". All other text in the URI is a literal string to be used as written. As explained in [RFC 2606] and [RFC 6761], the domains example.com, example.org and example.net are second-level domain names reserved by the Internet Assigned Numbers Authority (IANA) for use in documentation. These should be understood as a placeholder for any registered second-level domain name.

2 Conformance to GS1 Digital Link

This section is normative

The GS1 Digital Link standard comprises a number of discrete documents against which conformance can be asserted. The core of this standard, GS1 Digital Link URI syntax, is expressed using ABNF grammar [RFC 5234] in section *4* such that conformance can be determined with certainty.

There is no single conformance statement for the entirety of GS1 Digital Link. It is therefore inappropriate to make a formal claim of broad conformance without citing the specific standard with which conformance is claimed.

It is worth noting that a GS1 Digital Link URI, like any Web URI or URL, does not have any intrinsic meaning. It may be treated in exactly the same way as any URL. It is only if it is parsed by a GS1-aware system that GS1 Aapplication Iidentifiers and their values can be extracted and processed. Examples of such systems include scanners that may treat a GS1 Digital Link URI as an alternative syntax to element strings, and conformant GS1 resolvers. Applications SHALL NOT assume that a URL that follows the syntax defined in this standard will point to a resolver. One way to test whether a Web URI does or does not point to a GS1 conformant resolver is to check for the presence of a Resolver Description File in the relevant Well-Known location /.well-known/gs1resolver [RFC 8615]. Details of the Resolver Description File are defined in GS1 Digital Link Standard: Resolution [DL-Resolution].

NOTE: This standard discusses complete URIs encoded in data carriers such as QR codes, Data Matrix codes and NFC tags. The potential use of software to construct those URIs from components discovered through scans of, for example, UPC/EAN barcodes or GS1 DataMatrix symbols, is out of scope.

3 What is a URI?

This section is informative

This section provides some clarification about what a Uniform Resource Identifier (URI) is, how URIs relate to Uniform Resource Names (URNs) and Uniform Resource Locators (URLs), as well as providing an explanation of the main structural elements within a Web URI.

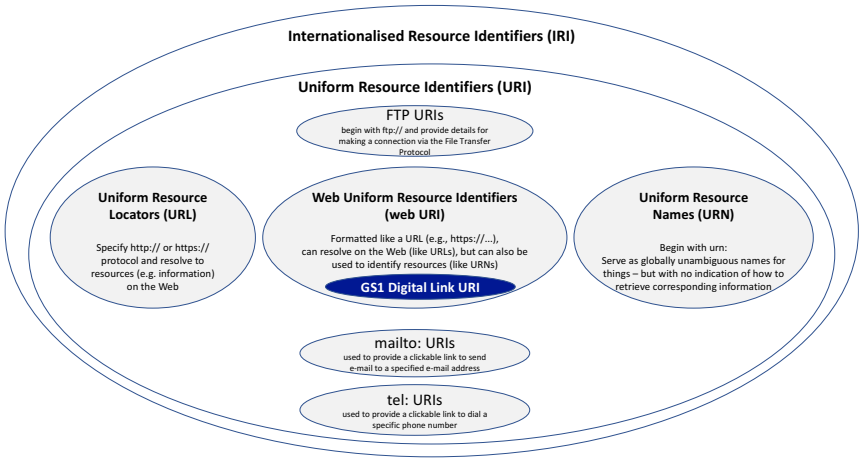

Figure 3-1 URNs and URLs are also URIs

Figure 3-1 shows a Venn diagram in which we see that Uniform Resource Identifier is the broad term that includes Uniform Resource Names (URNs) and Uniform Resource Locators (URLs) as well as URIs with various protocols including http or https, ftp, mailto, tel etc. This means that every URL and every URN is also a URI, since URI is the broader umbrella term. Furthermore, Internationalized Resource Identifiers (IRIs) are an even broader category that support characters from the Universal Character Set/Unicode, whereas URIs only support the ASCII character set. IRIs are defined in [IRIs]. GS1 Digital Link URIs are a subset of Web URIs that conform to this GS1 technical standard.

Figure 3-2 shows another Venn diagram. This time, it shows two capabilities:

1. The capability to easily resolve to resources (e.g. information) on the Web.
2. The capability to provide a globally unambiguous name for anything, whether or not the thing exists only on the Web or in the real world.

The first capability is usually associated with URLs and Web addresses.

The second capability is usually associated with URNs.

Web URIs exist at the intersection of these two capabilities; in terms of their syntax, they look like URLs because they specify http or https as their protocol - and they can be configured to behave like URLs in terms of supporting Web requests via the http/https Web protocol. However, they are also a perfectly valid way of assigning a globally unambiguous name for anything, whether in the real world or online. Note that 'globally unambiguous' does not mean globally unique; two different things should have distinct URIs in any situation where we want to be able to distinguish between them. However, there may be many URIs that all refer to the same thing, even within the same URI namespace or domain name. It is also possible to use Linked Data [Linked Data] to make an

assertion between two URIs to formally express that they both refer to the same thing, even if the URIs are different strings.

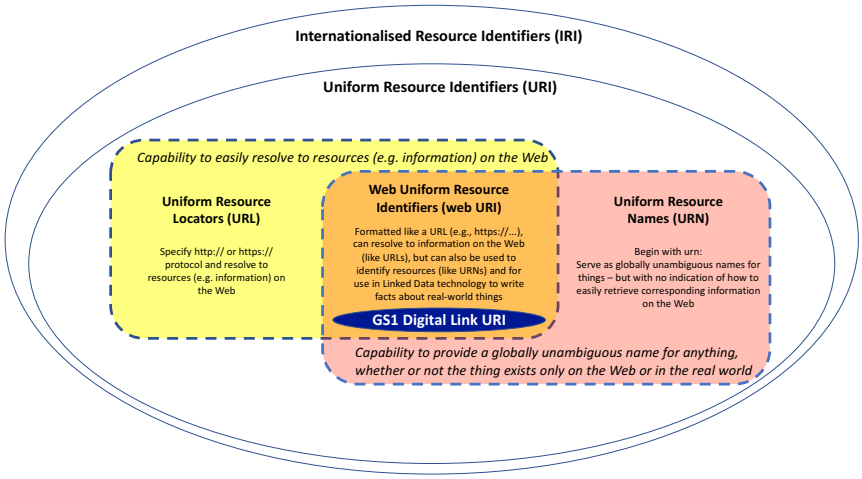

Figure 3-2 A Web URI can act both as a globally unambiguous name for something, as well as providing an easy way to retrieve Web resources (e.g. information) relating to the identified thing

Figure 3-3 provides a brief overview of the internal structural elements of a Web URI:

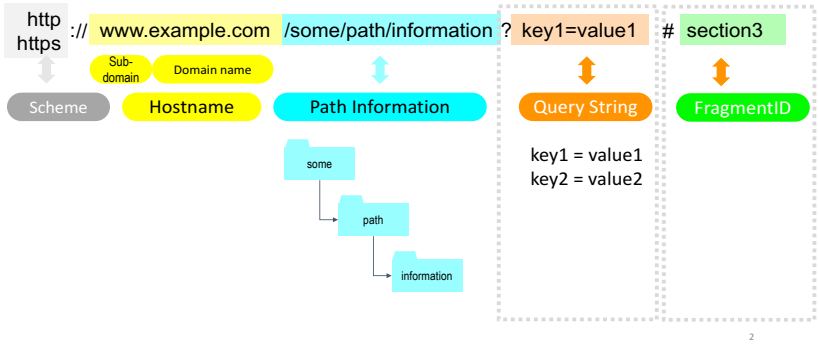

Figure 3-3 Internal structure of a Web URI

Figure 3-3 shows the structural elements of a Web URI. The scheme indicates the protocol and (at the time of writing) is always http:// or https:// (use of HTTPS is more secure and is therefore recommended as best practice). The hostname is typically a registered Internet domain name or a subdomain of such a registered domain name. Following the domain name, the remainder of the Web URI is case sensitive. The URI path information consists of a number of strings separated by the forward slash character. Although this is just a string, it is often used by the Linked Data

community and in REST interfaces [REST] to represent a collection of resources organised in a conceptually hierarchical way, with the broadest (most general, least specific) category appearing towards the left of the URI path information and with the narrowest (most specific) category appearing towards the right of the URI path information.

This design pattern provides a hint to humans that related Web URIs may exist and can be formed by successively truncating the Web URI path information from right to left, removing each successive segment preceded by its forward slash ("/") character. These related Web URIs may provide information about an object at a broader, more general, less specific granularity.

However, this is only a legible hint to humans. Computer software would typically treat the entire URI (at least up to the fragment identifier) as an opaque indivisible string and would not attempt such truncation. Instead, they will look for explicit links to related URIs, ideally expressed with semantic annotation, using Linked Data properties. These aspects – the machine-processable semantics or meaning of a GS1 Digital Link URI – are explored and defined in detail in GS1 Digital Link Standard: Semantics [DL-Semantics]

The query string enables multiple key=value pairs to be sent to a Web resource. The URI query string appears after the URI path information and consists of everything between the "?" at the end of the path information and the end of the URI or the "#" symbol indicating the start of the fragment identifier. Within the URI query string, key=value pairs may be concatenated using & or ; as a delimiter.

The URI fragment identifier is optional and appears after the query string (if present) and preceded by the "#" character. The URI fragment identifier is typically used to provide a link to an internal subsection of an information resource. The Linked Data community do make use of URIs with fragment identifiers, although the fragment identifier is not useful for passing key=value pairs. Importantly, fragment identifiers are *not* sent to the server but are handled entirely within the client.

Web URIs provide essentially two options for expressing the values of GS1 Application Identifiers - either within the URI path information or within the URI query string. The URI path information is the most appropriate place for expressing a GS1 identification key and an ordered set of optional qualifiers that are used in conjunction with the GS1 identification key to form a compound key that is used to retrieve information about something at a finer level of granularity (e.g. traceability data about an SGTIN, batch/lot-level master data). The query string is appropriate for data attributes of the identified resource such as expiry date, weight etc., as well as being a natural extension point for any additional arbitrary key=value pairs that cannot be expressed using GS1 Application Identifiers (see section *4.10.1*); for example, the query string could include a key=value pair to indicate a specific stakeholder role or a specific action or activity or type of service to be accessed. It should be noted that no key=value pair should be repeated with the same key in the URI query string. If a key is repeated, the last defined value for that key takes precedence over any previously defined value.

3.1 The GS1 Digital Link URI

GS1 Digital Link provides a syntax for expressing GS1 identifier keys, key qualifiers and data attributes in a format that can be used on the Web in an intuitive manner (via a straightforward HTTP request) to enable consumers and others to directly access relevant information and services about products, assets, locations, etc. A GS1 Digital Link URI can be encoded natively in any data carrier that can support the encoding of a Web address (URL). This means that additional data carriers such as QR Codes®, digital watermarks, NFC tags and other technologies will also be able to include GS1 identification keys while continuing to provide links to relevant information. When the data carrier is created and such a URL is embedded within it, a scanning device can extract the entire URL, and no further processing by the scanning device, or software therein, is required to construct the URL that is used to access a server where relevant information is stored.

4 GS1 Digital Link URI Syntax

This section and all its subsections are normative

This section specifies the structure of GS1 Digital Link URIs using the Augmented Backus-Naur Form (ABNF) syntax as defined in [RFC 5234] and updated by [RFC 7405]. ABNF formally expresses how strings of characters (including URIs) are constructed by concatenating smaller components in a sequential order and is machine-processable.

Those smaller components may be defined in terms of further sub-components and/or in terms of sequences of character sets that are also defined by rules.

ABNF also supports repeating components and optional components. Optional components are enclosed within square brackets.

A sequential group of one or more components may be enclosed within round brackets.

Repeating components use the `m*n(component)` notation to indicate that the component within the round brackets may appear at least `m` times and at most `n` times. Default values are `m`=0, `n`=infinity. If either or `m` or `n` are omitted, their default values are assumed.

Everything following a semicolon on a line is considered to be an explanatory comment.

The notation `n(component)` or `n`*component* where `n` is one or more digit characters is equivalent to `n*n(component)`, indicating that the component must appear exactly n times.
A number of comments are provided to explain the meaning of rules.

ABNF is designed primarily to express formal syntax in standards documents. It may also be used to validate strings against that syntax, however, there are limitations. It has no negation option (string SHALL NOT contain "xyz") and it does not support non-greedy matching. For this reason, there are some features of the GS1 Digital Link URI syntax that cannot be tested using ABNF-based parsers. In particular, those with a custom path will fail ABNF-based validation.

4.1 ~~Deprecation warning~~Convenience alphas removed

Earlier versions of t~~T~~he formal grammar below, developed initially for the first version of the GS1 Digital Link standard [DL1], supporteds~~‘convenience alphas’~~ in place of commonly used GS1 ~~application~~ GS1 Application Iidentifiers. For example, ‘01’ couldan~~be replaced by ‘gtin’, ‘414’ by ‘gln’~~ etc. These were introduced in an effort to make GS1 Digital Link URIs more developer-friendly. Experience has shown that the opposite is true as it introduceds~~complexity, and confusion and inconsistency~~ for implementations of the standard. At the time of writing, there are many implementations of GS1 Digital Link by scanning equipment manufacturers and barcode generating tools. The majority of these do not recognise convenience alphas and the small number of known implementations that do are being updated to remove them. ~~Therefor~~Convenience alphas~~ were marked as deprecated in version 1.2 of the standard and have been removed completely as of version 1.2.23.0. please note that:~~

~~1. Convenience alphas will be removed from future versions of the standard and hence are flagged as DEPRECATED here.~~

~~This is reflected as relevant in later sections of this document.~~

4.2 Character sets

Firstly, a number of character sets are defined for later re-use in subsequent ABNF rules.

```
DIGIT      = "0" / "1" / "2" / "3" / "4" /
             "5" / "6" / "7" / "8" / "9"

BOOLEAN    = "0" / "1"


UPPERALPHA = %x41-5A ; A-Z  ( ASCII characters 65-90 decimal, 41-5A hex)
```

```
LOWERALPHA = %x61-7A ; a-z  ( ASCII characters 97-122 decimal, 61-7A hex)

ALPHA      = UPPERALPHA / LOWERALPHA ; A-Z or a-z

HEXDIG     = DIGIT / "A" / "B" / "C" / "D" / "E" / "F"

DoubleQuote = '"' ; the double-quote character "
```

The following characters must be represented using percent-encoding (see section 2.1 of RFC 3986 [PercentEncoding]) when used as literal characters within URIs, since many of these have special meanings within Web URIs:

```
Octothorpe           = "%23" ; percent-encoding of the # character
ForwardSlash         = "%2F" ; percent-encoding of the / character

Percent              = "%25" ; percent-encoding of the % character
Ampersand            = "%26" ; percent-encoding of the & character
Plus                 = "%2B" ; percent-encoding of the + character
Comma                = "%2C" ; percent-encoding of the , character

Exclamation          = "%21" ; percent-encoding of the ! character
LeftBracket          = "%28" ; percent-encoding of the ( character
RightBracket         = "%29" ; percent-encoding of the ) character
Asterisk             = "%2A" ; percent-encoding of the * character

Apostrophe           = "%27" ; percent-encoding of the ' character
Colon                = "%3A" ; percent-encoding of the : character
Semicolon            = "%3B" ; percent-encoding of the ; character
LeftAngleBracket     = "%3C" ; percent-encoding of the < character
Equals               = "%3D" ; percent-encoding of the = character
RightAngleBracket    = "%3E" ; percent-encoding of the > character
QuestionMark         = "%3F" ; percent-encoding of the ? character
```

The following group of symbol characters is permitted within the 82-character subset of ISO/IEC 646, indicated in Figure 7.11-1 of the GS1 General Specifications [GENSPECS].

```
XSYMBOL          = DoubleQuote / "-" / "." / "_" / Exclamation / Percent /
                   Ampersand / Plus / Comma / ForwardSlash / Asterisk /
                   LeftBracket / RightBracket / Apostrophe / Semicolon /
                   Colon / LeftAngleBracket / RightAngleBracket / Equals /
                   QuestionMark
```

The following group of symbol characters is permitted within the 39-character subset of ISO/IEC 646, indicated in Figure 7.11-2 of the GS1 General Specifications [GENSPECS].

```
YSYMBOL          = "-" / Octothorpe / ForwardSlash
```

The following character set corresponds to all permitted characters within the 82-character subset of ISO/IEC 646, indicated in Figure 7.11-1 of the GS1 General Specifications [GENSPECS].

```
XCHAR            = DIGIT / UPPERALPHA / LOWERALPHA / XSYMBOL
```

The following character set corresponds to all permitted characters within the 39-character subset of ISO/IEC 646, indicated in Figure 7.11-2 of the GS1 General Specifications [GENSPECS]. It is currently only used within the value of the Components and Parts Identifier (CPID).

```
YCHAR            = DIGIT / UPPERALPHA / YSYMBOL
```

4.3 Primary identification keys

The following rules indicate which GS1 Application Identifiers (AI) that are considered as primary identification keys for GS1 Digital Link URI. Please note that as of version 1.3.02.2 of this standard, the 'convenience alphas' defined in earlier versions are no longer supported (see section 4.1). Note that for each of these (and the rules in section 4.4), the numeric AI value may be used or alternatively, a corresponding lower-case short name may be used if it is more friendly to software developers. The numeric AI value may be more suitable for use when encoding a GS1 Digital Link URI within a 2D barcode, since this can be encoded more efficiently, resulting in a lower total module count and improved readability.

The %s prefix notation was introduced in [RFC 7405] and simply indicates that the following string value is case-sensitive. For example, in the rule below, gtin-code may be either "01" or "gtin" but not "GTIN" nor "Gtin".

Note: the alphanumeric notation below will be deprecated and scheduled to be removed from future versions of the standard.

```
gtin-code        = "01" / %s"gtin"          ; GTIN
itip-code        = "8006" / %s"itip" ; ITIP
gmn-code         = "8013" / %s"gmn"  ; Global Model Number
cpid-code        = "8010" / %s"cpid" ; CPID
gln-code         = "414" / %s"gln"          ; Physical Location GLN
payTo-code       = "415"       ; GLN of invoicing party
partyGln-code    = "417" / %s"party"        ; Party GLN
gsrnp-code       = "8017" / %s"gsrnp"       ; GSRN of the Provider
gsrn-code        = "8018" / %s"gsrn" ; GSRN of the Recipient
gcn-code         = "255" / %s"gcn"          ; GCN
sscc-code        = "00" / %s"sscc"          ; SSCC
gdti-code        = "253" / %s"gdti"         ; GDTI
ginc-code        = "401" / %s"ginc"         ; GINC
gsin-code        = "402" / %s"gsin"         ; GSIN
grai-code        = "8003" / %s"grai" ; GRAI
giai-code        = "8004" / %s"giai" ; GIAI
```

Commented [PD1]: Removed, as it seems %s-prefix only relates to the alphas ?

Commented [PA2]: WR 22-255 covered by this one line (re-inserted from version 1.1 of the spec)

4.4 Key qualifiers

The following rules which GS1 Application Identifiers (AI) are considered as key qualifiers for a GS1 Digital Link URI.

```
cpv-code              = "22"      / %s"cpv"            ; Consumer Product
Variant
lot-code              = "10"      / %s"lot"      ; Batch/Lot identifier
ser-code              = "21"      / %s"ser"            ; GTIN Serial Number
cpsn-code             = "8011"      / %s"cpsn"  ; CPID Serial Number
glnx-code             = "254"       / %s"glnx"  ; GLN extension
refno-code            = "8020"      / %s"refno" ; Payment Reference Number
srin-code             = "8019"      / %s"srin"  ; Service Relation Instance
          Number
tpx-code              = "235"       ; third-party controlled serialised
                                      extension to GTIN
uic-ext-code          = "7040"   ; GS1 UIC with Extension 1 and Importer
                                   Index
```

4.5 Primary key formats

The following rules express the format of the values of the primary GS1 identification keys.

Note: the GS1 General Specifications [GENSPECS] define further restrictions on some of these values, particularly for those which include a GS1 Check Digit, Indicator Digit or Extension Digit. Please refer to the GS1 General Specifications [GENSPECS] for further details.

```
gtin-value            = 8DIGIT / 12DIGIT / 13DIGIT / 14DIGIT
itip-value            = 14DIGIT 2DIGIT 2DIGIT
                        ; 14 digits then 2 digits then 2 digits
gmn-value             = 1*25XCHAR  ; 1-25 characters from 82-chr subset
cpid-value            = 1*30YCHAR  ; 1-30 characters from 39-chr subset
gln-value             = 13DIGIT   ; exactly 13 digits
payTo-value           = 13DIGIT   ; exactly 13 digits
partyGln-value        = 13DIGIT   ; exactly 13 digits
gsrnp-value           = 18DIGIT   ; exactly 18 digits
gsrn-value            = 18DIGIT   ; exactly 18 digits
gcn-value             = 13DIGIT [1*12DIGIT]
                        ; 13 digits then optional 1-12 digits
sscc-value            = 18DIGIT   ; exactly 18 digits
gdti-value            = 13DIGIT [1*17XCHAR]
                        ; 13 digits then optional 1-17 characters
                        ; from the 82-character subset
ginc-value            = 1*30XCHAR
                        ; 1-30 characters from the 82-character subset
gsin-value            = 17DIGIT   ; exactly 17 digits
```

Commented [PA3]: Part of WR 22-255

```
grai-value          = 14DIGIT [1*16XCHAR]
                      ; 14 digits then optional 1-16 characters
                      ; from the 82-character subset of ISO/IEC 646
giai-value          = 1*30XCHAR  ; 1-30 characters from 82-chr subset
```

4.6 Key qualifier formats

The following rules express the format of the values of the key qualifiers of primary GS1 identification keys:

```
cpv-value           = 1*20XCHAR  ; 1-20 characters from 82-chr subset
lot-value           = 1*20XCHAR  ; 1-20 characters from 82-chr subset
ser-value           = 1*20XCHAR  ; 1-20 characters from 82-chr subset
cpsn-value          = 1*12DIGIT  ; 1-12 digits
glnx-value          = 1*20XCHAR  ; 1-20 characters from 82-chr subset
refno-value         = 1*25XCHAR  ; 1-25 characters from 82-chr subset
srin-value          = 1*10DIGIT  ; 1-10 digits
tpx-value           = 1*28XCHAR  ; 1-28 characters from 82-chr subset
uic-ext-value       = 1DIGIT 3XCHAR
                    ; 1 digit then 3 characters from 82-chr subset
```

4.7 Primary identifier and value concatenation

The following rules express how each primary identifier code and its value should be concatenated (for use within the URI path information) :

```
gtin-comp           = "/" gtin-code "/" gtin-value
itip-comp           = "/" itip-code "/" itip-value
gmn-comp            = "/" gmn-code "/" gmn-value
cpid-comp           = "/" cpid-code "/" cpid-value
gln-comp            = "/" gln-code "/" gln-value
payTo-comp          = "/" payTo-code "/" payTo-value
partyGln-comp       = "/" partyGln-code "/" partyGln-value
gsrnp-comp          = "/" gsrnp-code "/" gsrnp-value
gsrn-comp           = "/" gsrn-code "/" gsrn-value
gcn-comp            = "/" gcn-code "/" gcn-value
sscc-comp           = "/" sscc-code "/" sscc-value
gdti-comp           = "/" gdti-code "/" gdti-value
ginc-comp           = "/" ginc-code "/" ginc-value
gsin-comp           = "/" gsin-code "/" gsin-value
grai-comp           = "/" grai-code "/" grai-value
giai-comp           = "/" giai-code "/" giai-value
```

Commented [PA4]: WR 22-255

4.8 Key qualifier concatenation

The following rules express how each key qualifier and its value should be concatenated (for use within the URI path information) :

```
cpv-comp            = "/" cpv-code "/" cpv-value
lot-comp            = "/" lot-code "/" lot-value
ser-comp            = "/" ser-code "/" ser-value
cpsn-comp           = "/" cpsn-code "/" cpsn-value
glnx-comp           = "/" glnx-code "/" glnx-value
refno-comp          = "/" refno-code "/" refno-value
srin-comp           = "/" srin-code "/" srin-value
tpx-comp            = "/" tpx-code "/" tpx-value
uic-ext-comp        = "/" uic-ext-code "/" uic-ext-value
```

4.9 Path element order

The following rules express how the URI path information should be structured for each primary GS1 identification key. Note that some primary identifiers such as SSCC do not have any associated key qualifier. Other primary identifiers such as GTIN may have multiple key qualifiers. The square bracket notation indicates that the enclosed key qualifier component may be omitted but the sequence in which they appear is important and must be preserved. For example, the rule for gtin-path would permit any of these:

```
/01/9520123456788/22/2A/10/ABC123/21/12345XYZ
/01/9520123456788/10/ABC123/
/01/9520123456788/10/ABC123/21/12345XYZ
/01/9520123456788/21/12345XYZ
```

but does not permit strings such as:

```
/01/9520123456788/21/12345XYZ/10/ABC123
```

in which the sequential ordering of the key qualifier components is not preserved.

```
gtin-path           = gtin-comp  [cpv-comp] [lot-comp] [ser-comp]
itip-path           = itip-comp  [cpv-comp] [lot-comp] [ser-comp]
gmn-path            = gmn-comp
cpid-path           = cpid-comp  [cpsn-comp]
gln-path            = gln-comp   [glnx-comp]
payTo-path          = payTo-comp
partyGln-path       = partyGln-comp
gsrnp-path          = gsrnp-comp [srin-comp]
gsrn-path           = gsrn-comp  [srin-comp]
gcn-path            = gcn-comp
sscc-path           = sscc-comp
gdti-path           = gdti-comp
ginc-path           = ginc-comp
gsin-path           = gsin-comp
grai-path           = grai-comp
giai-path           = giai-comp
upui-path           = gtin-comp tpx-comp
eoid-path           = partyGln-comp uic-ext-comp
fid-path            = gln-comp uic-ext-comp
mid-path            = giai-comp uic-ext-comp
```

Commented [PD5]: WR 22-255

The following rule simply states that any of the above is considered as a gs1path (which will be referenced in a later rule).

```
gs1path    =     gtin-path / itip-path / gmn-path / cpid-path / gln-path /
                 payTo-path / partyGln-path / gsrnp-path / gsrn-path /
                 gcn-path /
                 sscc-path / gdti-path / ginc-path / gsin-path / grai-path
                 / giai-path / upui-path / eoid-path / fid-path / mid-path
```

Commented [PD6]: WR-255

4.10 Data attributes

The following rules are concerned with GS1 Application Identifiers that are considered to be data attributes rather than primary identifier keys or key qualifiers. Data attributes and their values SHALL be expressed via the URI query string as key=value pairs. Where there is a choice, the numeric AI value is much preferred over the more human-friendly short name.

Note that 'data attributes' MAY include AIs that may also be used as primary keys. In any GS1 Digital Link URI there SHALL be exactly one primary key, as defined in section 4.3, followed by any key qualifiers relevant to that primary key as path elements. However, the GS1 General Specifications [GENSPECS] allow combinations of primary keys in a single data carrier. For example, it is possible to encode both a GTIN and a GIAI in a single element string within a data carrier (see the example in section 5.11). Where it is necessary to encode more than one primary key in a single GS1 Digital Link URI, one SHALL be used in the path and the remaining key(s) encoded in the query string as data attributes.

```
netWeightVMTICode   = "3100" / "3101" / "3102" / "3103" / "3104" / "3105" /
                      "3200" / "3201" / "3202" / "3203" / "3204" / "3205" /
                      "3560" / "3561" / "3562" / "3563" / "3564" / "3565" /
                      "3570" / "3571" / "3572" / "3573" / "3574" / "3575"
netWeightVMTIValue        = 6DIGIT
netWeightVMTIParameter = netWeightVMTICode "=" netWeightVMTIValue

lengthVMTICode = "3110" / "3111" / "3112" / "3113" / "3114" / "3115" /
                 "3210" / "3211" / "3212" / "3213" / "3214" / "3215" /
                 "3220" / "3221" / "3222" / "3223" / "3224" / "3225" /
                 "3230" / "3231" / "3232" / "3233" / "3234" / "3235"
lengthVMTIValue     = 6DIGIT
lengthVMTIParameter = lengthVMTICode "=" lengthVMTIValue

widthVMTICode = "3120" / "3121" / "3122" / "3123" / "3124" / "3125" /
                "3240" / "3241" / "3242" / "3243" / "3244" / "3245" /
                "3250" / "3251" / "3252" / "3253" / "3254" / "3255" /
                "3260" / "3261" / "3262" / "3263" / "3264" / "3265"
widthVMTIValue     = 6DIGIT
widthVMTIParameter = widthVMTICode "=" widthVMTIValue

depthVMTICode = "3130" / "3131" / "3132" / "3133" / "3134" / "3135" /
                "3270" / "3271" / "3272" / "3273" / "3274" / "3275" /
                "3280" / "3281" / "3282" / "3283" / "3284" / "3285" /
                "3290" / "3291" / "3292" / "3293" / "3294" / "3295"
depthVMTIValue     = 6DIGIT
depthVMTIParameter = depthVMTICode "=" depthVMTIValue

areaVMTICode = "3140" / "3141" / "3142" / "3143" / "3144" / "3145" /
               "3500" / "3501" / "3502" / "3503" / "3504" / "3505" /
               "3510" / "3511" / "3512" / "3513" / "3514" / "3515" /
               "3520" / "3521" / "3522" / "3523" / "3524" / "3525"
areaVMTIValue     = 6DIGIT
areaVMTIParameter = areaVMTICode "=" areaVMTIValue
```

```
587         netVolumeVMTICode = "3150" / "3151" / "3152" / "3153" / "3154" / "3155" /
588                             "3160" / "3161" / "3162" / "3163" / "3164" / "3165" /
589                             "3600" / "3601" / "3602" / "3603" / "3604" / "3605" /
590                             "3610" / "3611" / "3612" / "3613" / "3614" / "3615" /
591                             "3640" / "3641" / "3642" / "3643" / "3644" / "3645" /
592                             "3650" / "3651" / "3652" / "3653" / "3654" / "3655" /
593                             "3660" / "3661" / "3662" / "3663" / "3664" / "3665"
594         netVolumeVMTIValue     = 6DIGIT
595         netVolumeVMTIParameter = netVolumeVMTICode "=" netVolumeVMTIValue
596
597
598         massPerUnitAreaVMTICode      = "3370" / "3371" / "3372" / "3373" / "3374" /
599                                        "3375"
600         massPerUnitAreaVMTIValue     = 6DIGIT
601         massPerUnitAreaVMTIParameter = massPerUnitAreaVMTICode "="
602                                                massPerUnitAreaVMTIValue
603
604
605         grossWeightCode   = "3300" / "3301" / "3302" / "3303" / "3304" / "3305" /
606                             "3400" / "3401" / "3402" / "3403" / "3404" / "3405"
607         grossWeightValue     = 6DIGIT
608         grossWeightParameter = grossWeightCode "=" grossWeightValue
609
610         logisticLengthCode = "3310" / "3311" / "3312" / "3313" / "3314" / "3315" /
611                              "3410" / "3411" / "3412" / "3413" / "3414" / "3415" /
612                              "3420" / "3421" / "3422" / "3423" / "3424" / "3425" /
613                              "3430" / "3431" / "3432" / "3433" / "3434" / "3435"
614         logisticLengthValue     = 6DIGIT
615         logisticLengthParameter = logisticLengthCode "=" logisticLengthValue
616
617         logisticWidthCode = "3320" / "3321" / "3322" / "3323" / "3324" / "3325" /
618                             "3440" / "3441" / "3442" / "3443" / "3444" / "3445" /
619                             "3450" / "3451" / "3452" / "3453" / "3454" / "3455" /
620                             "3460" / "3461" / "3462" / "3463" / "3464" / "3465"
621         logisticWidthValue       = 6DIGIT
622         logisticWidthParameter   = logisticWidthCode "=" logisticWidthValue
623
624         logisticDepthCode = "3330" / "3331" / "3332" / "3333" / "3334" / "3335" /
625                             "3470" / "3471" / "3472" / "3473" / "3474" / "3475" /
626                             "3480" / "3481" / "3482" / "3483" / "3484" / "3485" /
627                             "3490" / "3491" / "3492" / "3493" / "3494" / "3495"
628         logisticDepthValue       = 6DIGIT
629         logisticDepthParameter   = logisticDepthCode "=" logisticDepthValue
630
631         logisticAreaCode = "3340" / "3341" / "3342" / "3343" / "3344" / "3345" /
632                            "3530" / "3531" / "3532" / "3533" / "3534" / "3535" /
633                            "3540" / "3541" / "3542" / "3543" / "3544" / "3545" /
634                            "3550" / "3551" / "3552" / "3553" / "3554" / "3555"
635         logisticAreaValue        = 6DIGIT
636         logisticAreaParameter    = logisticAreaCode "=" logisticAreaValue
637
638         logisticVolumeCode = "3350" / "3351" / "3352" / "3353" / "3354" / "3355" /
639                              "3360" / "3361" / "3362" / "3363" / "3364" / "3365" /
640                              "3620" / "3621" / "3622" / "3623" / "3624" / "3625" /
641                              "3630" / "3631" / "3632" / "3633" / "3634" / "3635" /
642                              "3670" / "3671" / "3672" / "3673" / "3674" / "3675" /
643                              "3680" / "3681" / "3682" / "3683" / "3684" / "3685" /
644                              "3690" / "3691" / "3692" / "3693" / "3694" / "3695"
645         logisticVolumeValue      = 6DIGIT
646         logisticVolumeParameter  = logisticVolumeCode "=" logisticVolumeValue
```

```
647
648         processorCode = "7030" / "7031" / "7032" / "7033" / "7034" / "7035" /
649                         "7036" / "7037" / "7038" / "7039"
650         processorValue      = 3DIGIT 1*27XCHAR
651         processorParameter  = processorCode "=" processorValue
652
653         contentParameter          = "02=" 14DIGIT
654
655         prodDateParameter         = "11=" 6DIGIT
656
657         dueDateParameter          = "12=" 6DIGIT
658
659         packDateParameter         = "13=" 6DIGIT
660
661         bestBeforeDateParameter   = "15=" 6DIGIT
662
663         sellByDateParameter       = "16=" 6DIGIT
664
665         firstFreezeDateParameter  = "7006=" 6DIGIT
666
667         harvestDateParameter      = "7007=" 6DIGIT [6DIGIT]
668
669         pricePerUnitParameter     = "8005=" 6DIGIT
670
671         variantParameter          = "20=" 2DIGIT
672
673         varCountParameter         = "30=" 1*8DIGIT
674
675         countParameter            = "37=" 1*8DIGIT
676
677         mutualParameter           = "90=" 1*30XCHAR
678
679         additionalIdParameter     = "240=" 1*30XCHAR
680
681         custPartNoParameter       = "241=" 1*30XCHAR
682
683         mtoVariantParameter       = "242=" 1*6DIGIT
684
685         pcnParameter              = "243=" 1*20XCHAR
686
687         secondarySerialParameter  = "250=" 1*30XCHAR
688
689         refToSourceParameter      = "251=" 1*30XCHAR
690
691         amountCode                = "3900" / "3901" / "3902" / "3903" / "3904" /
692                                     "3905"
693
694         amountValue               = 1*15DIGIT
695
696         amountParameter           = amountCode "=" amountValue
697
698         amountISOCode             = "3910" / "3911" / "3912" / "3913" / "3914" /
699                                     "3915"
700         amountISOValue            = 3DIGIT 1*15DIGIT
701
702         amountISOParameter        = amountISOCode "=" amountISOValue
703
704         priceCode                 = "3920" / "3921" / "3922" / "3923" / "3924" /
705                                     "3925"
706         priceValue                = 1*15DIGIT
```

```
priceParameter            = priceCode "=" priceValue

priceISOCode              = "3930" / "3931" / "3932" / "3933" / "3934" /
                            "3935"
priceISOValue             = 3DIGIT  1*15DIGIT

priceISOParameter         = priceISOCode "=" priceISOValue

percentOffCode            = "3940" / "3941" / "3942" / "3943" / "3944" /
                            "3945"

percentOffValue           = 4DIGIT

percentOffParameter       = percentOffCode "=" percentOffValue

orderNumberParameter      = "400=" 1*30XCHAR

routeParameter            = "403=" 1*30XCHAR

shipToLocParameter        = "410=" 13DIGIT

billToParameter           = "411=" 13DIGIT

purchaseFromParameter     = "412=" 13DIGIT

shipForLocParameter       = "413=" 13DIGIT

payToParameter            = "415=" 13DIGIT

prodServLocParameter      = "416=" 13DIGIT

shipToPostParameter       = "420=" 1*20XCHAR

shipToPostISOParameter    = "421=" 3DIGIT 1*9XCHAR

originParameter           = "422=" 3DIGIT

countryProcessParameter   = "424=" 3DIGIT

countryFullProcessParameter  = "426=" 3DIGIT

countryInitialProcessParameter     = "423=" 3DIGIT 1*12DIGIT

countryDisassemblyParameter  = "425=" 3DIGIT 1*12DIGIT

originSubdivisionParameter   = "427=" 1*3XCHAR

nhrnPZNParameter          = "710=" 1*20XCHAR

nhrnCIPParameter          = "711=" 1*20XCHAR

nhrnCNParameter           = "712=" 1*20XCHAR

nhrnDRNParameter          = "713=" 1*20XCHAR

nhrnAIMParameter          = "714=" 1*20XCHAR

nhrnUS-FDAParameter       = "715=" 1*20XCHAR
```

Commented [PA7]: payToParameter deleted here as included with other primary keys as attributes a little further down in this section.

```
nsnParameter              = "7001=" 13DIGIT

meatCutParameter          = "7002=" 1*30XCHAR

activePotencyParameter    = "7004=" 1*4DIGIT

catchAreaParameter        = "7005=" 1*12XCHAR

aquaticSpeciesParameter   = "7008=" 1*3XCHAR

fishingGearTypeParameter  = "7009=" 1*10XCHAR

prodMethodParameter       = "7010=" 1*2XCHAR

refurbLotParameter        = "7020=" 1*20XCHAR

funcStatParameter         = "7021=" 1*20XCHAR

revStatParameter          = "7022=" 1*20XCHAR

giaiAssemblyParameter     = "7023=" 1*30XCHAR

certificationRefCode      = "7230" / "7231" / "7232" / "7233" / "7234" /
                            "7235" / "7236" / "7237" / "7238" / "7239"

certificationRefValue     = 2XCHAR 1*28XCHAR

certificationRefParameter = certificationRefCode "=" certificationRefValue

dimensionsParameter       = "8001=" 14DIGIT

cmtNoParameter            = "8002=" 1*20XCHAR

ibanParameter             = "8007=" 1*34XCHAR

prodTimeParameter         = "8008=" 8DIGIT [2DIGIT] [2DIGIT]

opticalSensorParameter    = "8009=" 1*50XCHAR

versionParameter          = "8012=" 1*20XCHAR

refNoParameter            = "8020=" 1*25XCHAR

itipContentParameter      = "8026=" 14DIGIT 2DIGIT 2DIGIT

couponIDNAParameter       = "8110=" 1*70XCHAR

pointsParameter           = "8111=" 4DIGIT

paperlessCouponIDNAParameter = "8112=" 1*70XCHAR

shipToCompParameter       = "4300=" 1*35XCHAR

shipToNameParameter       = "4301=" 1*35XCHAR

shipToAdd1Parameter       = "4302=" 1*70XCHAR

shipToAdd2Parameter       = "4303=" 1*70XCHAR

shipToSubParameter        = "4304=" 1*70XCHAR
```

```
827
828         shipToLocalityParameter   = "4305=" 1*70XCHAR
829
830         shipToRegParameter        = "4306=" 1*70XCHAR
831
832         shipToCountryParameter    = "4307=" 2XCHAR
833
834         shipToPhoneParameter      = "4308=" 1*30XCHAR
835
836         rtnToCompParameter        = "4310=" 1*35XCHAR
837
838         rtnToNameParameter        = "4311=" 1*35XCHAR
839
840         rtnToAdd1Parameter        = "4312=" 1*70XCHAR
841
842         rtnToAdd2Parameter        = "4313=" 1*70XCHAR
843
844         rtnToSubParameter         = "4314=" 1*70XCHAR
845
846         rtnToLocParameter         = "4315=" 1*70XCHAR
847
848         rtnToRegParameter         = "4316=" 1*70XCHAR
849
850         rtnToCountryParameter     = "4317=" 2XCHAR
851
852         rtnToPostParameter        = "4318=" 1*20XCHAR
853
854         rtnToPhoneParameter       = "4319=" 1*30XCHAR
855
856         srvDescriptionParameter   = "4320=" 1*35XCHAR
857
858         dangerousGoodsParameter   = "4321=" BOOLEAN
859
860         authToLeaveParameter      = "4322=" BOOLEAN
861
862         sigRequiredParameter      = "4323=" BOOLEAN
863
864         notBeforeDelDateParameter = "4324=" 10DIGIT
865
866         notAfterDelDateParameter  = "4325=" 10DIGIT
867
868         releaseDateParameter      = "4326=" 6DIGIT
869
870         amountPayPerUnitCode      = "3950" / "3951" / "3952" / "3953"
871         amountPayPerUnitValue     = 6DIGIT
872         amountPayPerUnitParameter = amountPayPerUnitCode "=" amountPayPerUnitValue
873
874         gtinParameter             = "01=" gtin-value
875
876         itipParameter             = "8006=" itip-value
877
878         gmnParameter              = "8013=" gmn-value
879
880         cpidParameter             = "8010=" cpid-value
881
882         glnParameter              = "414=" gln-value
883
884         payToParameter            = "415=" payTo-value
885
886         partyGlnParameter         = "417=" partyGln-value
```

Commented [PA8]: WR 22-255

```
gsrnpParameter            = "8017=" gsrnp-value

gsrnParameter             = "8018=" gsrn-value

gcnParameter              = "255=" gcn-value

ssccParameter             = "00=" sscc-value

gdtiParameter             = "253=" gdti-value

gincParameter             = "401=" ginc-value

gsinParameter             = "402=" gsin-value

graiParameter             = "8003=" grai-value

giaiParameter             = "8004=" giai-value

internalCode              = "91" / "92" / "93" / "94" / "95" /
                            "96" / "97" / "98" / "99"
internalValue             = 1*90XCHAR
internalParameter         = internalCode "=" internalValue
```

Batch/Lot may also be used as a data attribute in conjunction with an SSCC [AI (00)] and a CONTENT [AI (02)] in order to indicate that the SSCC contains GTINs of a specific batch/lot. For this reason, `LotParameter` is defined for use within the URI query string.

```
LotParameter     = lot-code "=" lot-value
```

Expiry Date [AI (17)] and Expiry Date/Time [AI (7003)] are data attributes. However, because of their importance in managing stock rotation and checking for expired products, they initially were given convenience alphas. These have now been removed (see section 4.1).

```
expiryDateCode         = "17"~~ / %s"exp"~~
expiryDateValue        = 6DIGIT
expiryDateParameter    = expiryDateCode "=" expiryDateValue

expiryTimeCode         = "7003"~~ / %s"expdt"~~
expiryTimeValue        = 10DIGIT
expiryTimeParameter    = expiryTimeCode "=" expiryTimeValue
```

4.10.1 Extension mechanism and reserved keywords

The URI query string is a natural extension point within the syntax that can accommodate additional key=value pairs to express data attribute parameters that cannot be expressed using GS1 Application Identifiers. Examples of such usage may be to express a specific role, action, activity or type of service to be accessed. The following `extensionParameter` is based on the ABNF rule for query appearing in [RFC 3986] and serves as the main extension point for the GS1 Digital Link URI syntax. It permits multiple arbitrary key=value pairs to be included within the query string of a GS1 Digital Link URI. Any key=value pairs used for extension data SHALL NOT be all-numeric to avoid conflict with existing and future keys used for GS1 Application Identifiers either in terms of semantics or syntax~~; and SHALL NOT use the values `lot`, `exp`, `expdt`~~; nor should they be used to express a value (such as a value for net weight) if that value can be expressed using GS1 Application Identifiers as data attributes. As detailed in GS1 Digital Link Standard: Resolution [DL-Resolution], the keywords `linkType` and `context` are also reserved and SHALL NOT be used except as defined in those sections.

```
extensionKey = *( DIGIT ) everythingExceptDigitsAndEquals
               *( DIGIT / everythingExceptDigitsAndEquals)

extensionValue = *( DIGIT / everythingExceptDigitsAndEquals / "=")
```

```
extensionParameter = extensionKey "=" extensionValue

                   ; any other query string parameter permitted by RFC 3986
                   ; including additional arbitrary key=value pairs except as
                   ; restricted in the above paragraph
```

4.10.2 Constructing the query string

The following rule states that any of the above parameters for data attributes may appear as a query string parameter (`queryStringParam`), referenced later.

```
queryStringParam  = netWeightVMTIParameter / lengthVMTIParameter /
                    widthVMTIParameter / depthVMTIParameter / areaVMTIParameter /
                    netVolumeVMTIParameter / massPerUnitAreaVMTIParameter /
                    grossWeightParameter / logisticLengthParameter /
                    logisticWidthParameter / logisticDepthParameter /
                    logisticAreaParameter / logisticVolumeParameter /
                    processorParameter / LotParameter / expiryDateParameter /
                    expiryTimeParameter / contentParameter / prodDateParameter /
                    dueDateParameter / packDateParameter / bestBeforeDateParameter /
                    sellByDateParameter / firstFreezeDateParameter /
                    harvestDateParameter / pricePerUnitParameter / variantParameter /
                    varCountParameter / countParameter / amountPayPerUnitParameter /
                    additionalIdParameter / custPartNoParameter /
                    mtoVariantParameter / pcnParameter / secondarySerialParameter /
                    refToSourceParameter / amountParameter / amountISOParameter /
                    priceParameter / priceISOParameter / percentOffParameter /
                    orderNumberParameter / routeParameter / shipToLocParameter /
                    billToParameter / purchaseFromParameter / shipForLocParameter /
                    prodServLocParameter / shipToPostParameter /
                    shipToPostISOParameter / originParameter /
                    countryProcessParameter / countryFullProcessParameter /
                    countryInitialProcessParameter / countryDisassemblyParameter /
                    originSubdivisionParameter / nhrnPZNParameter /nhrnCIPParameter /
                    nhrnCNParameter / nhrnDRNParameter / nsnParameter /
                    meatCutParameter / activePotencyParameter / catchAreaParameter /
                    fishingGearTypeParameter / prodMethodParameter /
                    refurbLotParameter / funcStatParameter / revStatParameter /
                    giaiAssemblyParameter / dimensionsParameter / cmtNoParameter /
                    ibanParameter / prodTimeParameter / versionParameter /
                    refNoParameter / couponIDNAParameter / pointsParameter /
                    itipContentParameter / certificationRefParameter /
                    aquaticSpeciesParameter / opticalSensorParameter /
                    paperlessCouponIDNAParameter /
                    internalParameter / mutualParameter / extensionParameter /

                    shipToCompParameter / shipToNameParameter /
                    shipToaAdd1Parameter / shipToaAdd2Parameter /
                    shipToSubParameter / shipToLocalityParameter /
                    shipToRegParameter / shipToCountryParameter /
                    shipToPhoneParameter / rtnToCompParameter /
                    rtnToNameParameter / rtnToAdd1Parameter /
                    rtnToAdd2Parameter / rtnToSubParameter /
                    rtnToLocParameter / rtnToRegParameter /
                    rtnToCountryParameter / rtnToPostParameter /
                    rtnToPhoneParameter / srvDescriptionParameter /
                    dangerousGoodsParameter / authToLeaveParameter /
                    sigRequiredParameter / notBeforeDelDateParameter /
                    notAfterDelDateParameter / releaseDateParameter /

                    gtinParameter  / itipParameter / gmnParameter / cpidParameter /
                    glnParameter / payToParameter / partyGlnParameter /
                   gsrnpParameter /
                   gsrnParameter / gcnParameter / ssccParameter / gdtiParameter /
```

```
                    - gincParameter / gsinParameter / graiParameter / giaiParameter
```

4.11 Constructing the GS1 Digital Link URI

The following rules are derived from rules appearing in [RFC 3986] and are used for defining the general structure of a Web URI. These are particularly relevant for GS1 Digital Link URIs that are not under the id.gs1.org domain.

```
scheme                = "http" / "https"

unreserved            = ALPHA / DIGIT / "-" / "." / "_" / "~"

reserved              = gen-delims / sub-delims

pct-encoded           = "%" HEXDIG HEXDIG

gen-delims            = ":" / "/" / "?" / "#" / "[" / "]" / "@"

sub-delims            = "!" / "$" / "&" / "'" / "(" / ")" / "*" /
                        "+" / "," / ";" / "="

sub-delims-without-equals = "!" / "$" / "&" / "'" / "(" / ")" /
                            "*" / "+" / "," / ";"

everythingExceptDigitsAndEquals = unreserved / pct-encoded /
                                  sub-delims-without-equals /
                                  ":" / "@" / "/" / "?"

pchar                 = unreserved / pct-encoded / sub-delims / ":" / "@"

segment               = *pchar

reg-name              = *( unreserved / pct-encoded / sub-delims )

dec-octet             = DIGIT                 ; 0-9
                      / %x31-39 DIGIT         ; 10-99
                      / "1" 2DIGIT            ; 100-199
                      / "2" %x30-34 DIGIT     ; 200-249
                      / "25" %x30-35          ; 250-255

IPv4address           = dec-octet "." dec-octet "." dec-octet "." dec-octet

IPv6address           = 6( h16 ":" ) ls32
                      /  "::" 5( h16 ":" ) ls32
                      / [ h16 ] "::" 4( h16 ":" ) ls32
                      / [ *1( h16 ":" ) h16 ] "::" 3( h16 ":" ) ls32
                      / [ *2( h16 ":" ) h16 ] "::" 2( h16 ":" ) ls32
                      / [ *3( h16 ":" ) h16 ] "::"    h16 ":"   ls32
                      / [ *4( h16 ":" ) h16 ] "::"              ls32
                      / [ *5( h16 ":" ) h16 ] "::"              h16
                      / [ *6( h16 ":" ) h16 ] "::"

ls32                  = ( h16 ":" h16 ) / IPv4address
                      ; least-significant 32 bits of address

h16                   = 1*4HEXDIG
                      ; 16 bits of address represented in hexadecimal

IP-literal            = "[" ( IPv6address / IPvFuture  ) "]"
```

```
IPvFuture             = "v" 1*HEXDIG "." 1*( unreserved / sub-delims / ":" )

host                  = IP-literal / IPv4address / reg-name

port                  = *DIGIT

hostname              = host [ ":" port ]
```

Finally, the following four rules define the syntax of a reference GS1 Digital Link URI from the concatenation of previously defined components:

```
queryStringDelim      = "&" / ";"

queryStringComp       =
             "?" queryStringParam *( queryStringDelim queryStringParam)

uncompressedGS1webURIPattern       = gs1path [queryStringComp]

referenceGS1webURI    = "https://id.gs1.org" uncompressedGS1webURIPattern
```

The following rules define the syntax of a non-reference GS1 Digital Link URI from the concatenation of previously defined components. An example of usage of a non-reference GS1 Digital Link URI is when a company chooses to use their own registered Internet domain name to construct the Web URI but aligns with this specification for the format of the final part of the URI path information and query string. Note that zero or more path segments are permitted to appear after the hostname or domain name and before the start of the `gs1uriPattern` defined in this specification.

```
optionalPathSegment   = "/" segment

customURIstem         = scheme "://" hostname *optionalPathSegment

uncompressedCustomGS1webURI  = customURIstem uncompressedGS1webURIPattern
```

The formal ABNF syntax for the URI should be read in combination with the GS1 General Specifications [GENSPECS] to ensure appropriate usage of Application Identifiers that represent data attributes of identified things. In particular, section 4.14 of the GS1 General Specifications [GENSPECS] provides guidance about data relationships, including invalid pairs of element strings (see section 4.14.1) and mandatory associations of element strings (see section 4.14.2). In the GS1 General Specifications [GENSPECS], section 2 specifies which identifiers are used for an application, section 3 provides definitions for each Application Identifier, while section 4 explains the management rules for each GS1 identification key.

As previously mentioned, some GS1 primary identifier keys include GS1 check digits and some also include indicator digits or extension digits that are to be used for specific purposes. Section 7 of the GS1 General Specifications [GENSPECS] provides details of AIDC validation rules and section 7.2.7 explains the GS1 check digit calculation. Nothing in this GS1 specification changes the existing validation rules that apply to the values of GS1 Application Identifiers; this document only specifies how valid GS1 AI values shall be expressed in the GS1 Digital Link structure.

Any URI that conforms to the formal syntax as defined above and that respects the relevant rules specified in the GS1 General Specifications as cited is:

1. a valid Web URI that can be dereferenced on the Web without further processing;
2. a valid expression of one or more GS1 application identifiers and their values, informationally equivalent to the same data expressed in GS1 AI syntax.

4.12 Canonical GS1 Digital Link URIs

A GS1 Digital Link URI can be constructed in any domain name, may contain additional key=value pairs in the query string and so on. This flexibility is a deliberate feature of the standard to support

its use in as many scenarios as possible and to ensure brands can remain in control of the domains they use.

However, in some contexts (e.g., to support carriers that cannot embed a Web URI) it is necessary to identify a *single* well-known or default version of the GS1 Digital Link URI. This is defined in [RFC 6596] as the *canonical URI*. We define the canonical URI as follows:

- the scheme SHALL be HTTPS;
- the domain name SHALL be id.gs1.org;
- deprecated convenience string equivalents for AIs SHALL NOT be used;
- GTIN-8, GTIN-12 and GTIN-13 SHALL be padded to 14 digits
- the URI query string (if present) SHALL NOT contain any other key=value pairs except for keys that are GS1 application identifiers;
- key=value pairs, if present, should be sorted in lexical, not numeric, order of the key;
- for clarity, this means that the parameters defined in GS1 Digital Link Standard: Resolution [DL-Resolution], namely `linkType` and `context`, and their values, are not included in the canonical GS1 Digital Link.

Commented [PA9]: Keeping this sentence in for now

It follows that the canonical version of

http://example.com/01/9520123456788/22/2A?linkType=gs1:traceability&foo=bar

is

`https://id.gs1.org/01/09520123456788/22/2A`

Some further points of clarification:

- A *canonical* GS1 Digital Link URI, as defined in this section, is distinct from a *reference* GS1 Digital Link URI, which is defined as *any* valid GS1 Digital Link URI on the id.gs1.org domain. The other rules above do not apply to reference GS1 Digital Link URIs.
- Elsewhere in the GS1 Digital Link standards it is stated that a trailing slash, while not-conformant, should be tolerated by resolvers. That is, a resolver should tolerate https://example.com/8003/{grai}/ even though that final / character is not allowed by the formal ABNF. A canonical GS1 Digital Link URI SHALL NOT include a trailing slash.

~~2.1 Preferred representation~~

~~The syntax defined for GS1 Digital Link URIs in general allows commonly used numeric AIs to be substituted for short strings that ease human comprehension. Note that this is not allowed for the canonical GS1 Digital Link URI structure defined immediately above in section *4.12*. For example, /01/ and /gtin/ are synonymous, as are /21/ and /ser/ etc. These can be very useful when explaining the standard to a new audience. However, as noted in section *4.1*, these 'convenience alphas' will be deprecated in the next iteration of this standard. Therefore, in any production setting or new implementation, only the numeric versions SHALL be used. There are three primary reasons for this:~~

~~3. In some data carriers, fewer bits or modules are needed to encode numbers than letters;~~

~~4. In all cases, the 'convenience alpha' short name is longer than the numeric version;~~

~~5. GS1 Digital Link is based on GS1 AI based element strings which are all based on numeric AIs. Convenience alphas confuse this and add an extra burden to implementations.~~

5 Examples of GS1 Digital Link URIs

This section is informative

5.1 GTIN

`https://id.gs1.org/01/09520123456788`

is the canonical Digital Link URI for GTIN `9520123456788` equivalent to the following element string:

`(01)09520123456788`

The following are further valid GS1 non canonical Digital Link URIs for GTIN `9520123456788` using a custom domain name e.g. example.com instead of `id.gs1.org`

`https://brand.example.com/01/9520123456788`

`https://brand.example.com/some-extra/pathinfo/01/9520123456788`

If redirection information has been specified to GS1 by the corresponding licensee of that GTIN or the GS1 Company Prefix (for GTINs constructed from GS1 Company Prefixes), a GS1 Resolver that supports GS1 Digital Link URIs will be able to effectively redirect any requests for that GS1 Digital Link URI to a corresponding URL specified by the licensee.

5.2 GTIN + CPV

`https://id.gs1.org/01/09520123456788/22/2A`

Is the canonical GS1 Digital Link URIs for GTIN `9520123456788` combined with Consumer Product Variant `'2A'` and to the following element string:

`(01)09520123456788(22)2A`

The following are further valid non canonical GS1 Digital Link URIs for GTIN `9520123456788` combined with Consumer Product Variant `'2A'`.

`https://brand.example.com/01/9520123456788/22/2A`

`https://retailer.example.com/01/9520123456788/22/2A`

5.3 GTIN + Batch/Lot

`https://id.gs1.org/01/09520123456788/10/ABC123`

is the canonical GS1 Digital Link URIs for GTIN `9520123456788` combined with Batch/Lot `'ABC123'`and equivalent to the following element string:

`(01)09520123456788(10)ABC123`

The following are further non canonical valid GS1 Digital Link URIs for GTIN `9520123456788` combined with Batch/Lot `'ABC123'`

`https://brand.example.com/01/9520123456788/10/ABC123`

`https://retailer.example.com/01/9520123456788/10/ABC123`

5.4 GTIN + Serial Number (also known as SGTIN)

`https://id.gs1.org/01/09520123456788/21/12345`

is the canonical GS1 Digital Link URIs for GTIN `9520123456788` combined with Serial Number `'12345'` and equivalent to the following element string:

`(01)09520123456788(21)12345`

The following are further valid GS1 Digital Link URIs for GTIN 9520123456788 combined with Serial Number '12345'

https://brand.example.com/01/9520123456788/21/12345

https://retailer.example.com/01/9520123456788/21/12345

5.5 GTIN + Batch/Lot + Serial Number + Expiry Date

https://id.gs1.org/01/09520123456788/10/ABC1/21/12345?17=180426

is the canonical GS1 Digital Link URIs for GTIN 9520123456788 combined with Batch/Lot 'ABC1' and Serial Number '12345' and with an expiry date of 26th April 2018 equivalent to the following element strings:

(01)09520123456788(17)180426(10)ABC1(21)12345

The following is also a valid non canonical GS1 Digital Link URIs for GTIN 9520123456788 combined with Batch/Lot 'ABC1' and Serial Number '12345' and with an expiry date of 26th April 2018.

https://example.com/01/9520123456788/10/ABC1/21/12345?17=180426

5.6 GTIN + Net Weight

https://id.gs1.org/01/09520123456788?3103=000195

is the canonical GS1 Digital Link URIs for GTIN 9520123456788 combined with a net weight of 0.195 kg equivalent to the following element strings:

(01)09520123456788(3103)000195

The following is a further valid non canonical GS1 Digital Link URIs for GTIN 9520123456788 combined with a net weight of 0.195 kg :

https://example.com/01/9520123456788?3103=000195

5.7 GTIN + Net weight + Amount payable + Best before date

https://example.com/01/9520123456788?3103=000195&3922=0299&17=201225

This GS1 Digital Link URI includes three data attributes for the given GTIN, which can be in any order in the query string. The equivalent element string is

(01)09520123456788(3103)000195(3922)0299(17)201225

The following GS1 Digital Link URIs are also equivalent, but only the second is canonical as the data attributes have been arranged in the lexical order of the AIs.

https://id.gs1.org/01/9520123456788?3103=000195&3922=0299&17=201225

https://id.gs1.org/01/9520123456788?17=201225&3103=000195&3922=0299

5.8 SSCC

https://id.gs1.org/00/952012345678912345

is the canonical GS1 Digital Link URI for SSCC 952012345678912345 equivalent to the following element string:

(00)952012345678912345

The following is a further valid non canonical GS1 Digital Link URIs for SSCC 952012345678912345 :

https://example.com/00/952012345678912345

5.9 SSCC with specified Content, Count and Batch/Lot

https://id.gs1.org/00/952012345678912345?02=09520123456788&37=25&10=ABC12

is the canonical GS1 Digital Link URIs for SSCC 952012345678912345 containing a count [AI (37)] of 25 instances of Content [AI (02)] 09520123456788 having Batch/Lot identifier [AI (10)] 'ABC123' equivalent to the following element strings:

(00)952012345678912345(02)09520123456788(37)25(10)ABC123

The following is a further non canonical valid GS1 Digital Link URI for SSCC 106141412345678908 containing a count [AI (37)] of 25 instances of Content [AI (02)] 09520123456788 having Batch/Lot identifier [AI (10)] 'ABC123':

https://example.com/00/952012345678912345?02=09520123456788&37=25&10=ABC123

5.10 Physical location represented by a GLN or GLN + GLN Extension

https://id.gs1.org/414/9520123456788

is the canonical GS1 Digital Link URI for GLN 9520123456788 equivalent to the following element string:

(414)9520123456788

https://id.gs1.org/414/9520123456788/254/32a%2Fb

Is the canonical GS1 Digital Link URIs for GLN 9520123456788 combined with a GLN extension '32a/b'. Note that because the forward slash character has a special meaning within Web URIs, it is replaced with %2F, its percent encoding, when it is being used as a literal value, rather than as a URI path separator.

It is equivalent to the following element strings:

(414)9520123456788(254)32a/b

The following is also a valid but non canonical GS1 Digital Link URIs for GLN 9520123456788 :

https://example.com/414/9520123456788

The following is a further valid non canonical GS1 Digital Link URIs for GLN 9520123456788 combined with a GLN extension '32a/b' :

https://example.com/414~~gln~~/9520123456788/254/32a%2Fb

5.11 GIAI + GTIN

https://example.com/8004/95206141412345 67?01=9520123456788

https://example.com/01/9520123456788?8004=9520614141234567

Both of these GS1 Digital Link URIs express the same combination of GS1 Application Identifiers. However, they are not equivalent. For the first example, in which the GIAI appears in the URI path information, the issuer of that GIAI asset identifier is the authority, whereas for the second example in which the GTIN is in the URI path information, the licensee of the GTIN (typically the brand owner or manufacturer) is the authority for that GS1 Digital Link URI.

Although both identify an item with GIAI `9520614141234567` that is an instance of GTIN `9520123456788`, the choice of which identifier to place in the URI path information does matter for resolvers that have a policy of only permitting referral records specified by the respective licensee of the GS1 identification key appearing in the URI path information. It also makes a difference from a semantic perspective. The first example expresses that "this thing is an asset identified by this GIAI – and it is/was also a product identified by this GTIN." The second example expresses that this is a product and that it also carries this asset identifier. The second example is unlikely to be encoded by the manufacturer or brand owner in a data carrier except when an instance of a product is manufactured for a specific known customer/asset owner. Most mass-produced products are made to stock rather than made to order/bespoke. This is an example where there are two primary identifiers in a single GS1 Digital Link URI. The equivalent element string is

`(01)09520123456788(8004)9520614141234567`

Although there are no specific rules about which of the 'two primary keys' should go in the path and which in the query string, the order is likely to be determined by the context. In this example, the GIAI will be assigned by the owner of the item who purchased it from the manufacturer who assigned the GTIN. In this scenario, it is the owner who would create the GS1 Digital Link URI and therefore it is very likely to be the GIAI that goes in the path – the owner's primary – rather than the manufacturer. The presence of multiple primary keys has an effect on the semantics that can be inferred from the URI. See GS1 Digital Link Standard: Semantics for more on this topic [DL-Semantics].

6 AIDC Issues

This section is normative

The use of GS1 Digital Link URIs in data carriers is governed by the GS1 General Specifications [GENSPECS]. That document defines the full GS1 system from the semantics of individual Application Identifiers and their permitted values, through to data carrier positioning and human-readable information and much more besides.

The following subsections supplement the General Specifications as they pertain specifically to GS1 Digital Link.

6.1 Recognising a GS1 Digital Link URI

There is no special character that can be included in a data carrier to indicate that what follows is a GS1 Digital Link URI. This is because there is no special character in any data carrier to indicate that what follows is a URL – and GS1 Digital Link URIs are URLs. This is a deliberate and important design feature: a general purpose scanning application, such as a consumer's mobile device, can scan a GS1 Digital Link URI and treat it like any other URL.

Applications might, however, want to recognise a GS1 Digital Link URI and, for example, make use of the GS1 identifiers or execute specific queries against a resolver. Therefore, some processing is necessary by the scanner to determine whether the string of characters is or is not conformant to this and other GS1 standards.

A scanner working within the GS1 system that recognises GS1 Digital Link SHALL only pass on the scanned string if it has determined that it is *plausibly* a conformant GS1 Digital Link URI. It is not required to carry out a full validation, which is left to the receiving application.

We offer a method based on regular expression matches for making this determination but any method is acceptable. It does not give an absolute assurance that the string *is* a conformant GS1 Digital Link URI, rather it detects strings that are definitely not and plausibly are GS1 Digital Link URIs. This is in line with many scanning applications that will recognise the presence of indicator characters at the start of a barcode and act accordingly but will not process the scanned string further before passing it to a receiving application.

6.1.1 Matching an uncompressed GS1 Digital Link URI

The following regular expression will match a valid uncompressed GS1 Digital Link URI as defined in this specification. It is unlikely, although, not impossible, that it will match a URL that is not also a conformant GS1 Digital Link URI. Failure to match means it definitely is not an uncompressed GS1 Digital Link URI.

In addition to the GS1 Digital Link syntax, all regular expressions provided below support the inclusion of a user name and port number in the URL. These are rarely used in practice but are part of the formal URL syntax.

RE1:

```
^https?:(\/\/((([^\/?#]*)@)?([^\/?#:]*)(:([^\/?#]*))?))?([^?#]*)(((\/(01|8006|8013|8010|414|415|417|8017|8018|255|00|253|401|402|8003|8004)\/)(\d{4}[^\/]+)(\/[^/]+\/[^/]+)?[/]?(\?([^?\n]*))?(#([^\n]*))?))~~^https?:(\/\/((([^\/?#]*)@)?([^\/?#:]*)(:([^\/?#]*))?))?([^?#]*)(((\/(01|gtin|8006|itip|8013|gmn|8010|cpid|414|gln|417|party|8017|gsrnp|8018|gsrn|255|gcn|00|sscc|253|gdti|401|ginc|402|gsin|8003|grai|8004|giai)\/)(\d{4}[^\/]+)(\/[^/]+\/[^/]+)?[/]?(\?([^?\n]*))?(#([^\n]*))?))~~
```

Commented [PD10]: WR 255

Commented [PA11R10]: Good catch, thanks

~~Regular expression RE1 provided above recognises the convenience alphas ('gtin' as an alias for '01' etc.). As noted in section 4.1, these are deprecated and will be removed in the next version of this standard. Therefore we provide a modified version of the above regular expression, RE2, that does not recognise the convenience alphas.~~

~~RE2:~~

```
~~^https?:(\/\/((([^\/?#]*)@)?([^\/?#:]*)(:([^\/?#]*))?))?([^?#]*)(((\/(01|8006|8013|8010|414|417|8017|8018|255|00|253|401|402|8003|8004)\/)(\d{4}[^\/]+)(\/[^/]+\/[^/]+)?[/]?(\?([^?\n]*))?(#([^\n]*))?))~~
```

~~6.1.5~~6.1.2 Matching a compressed GS1 Digital Link URI

The following regular expression, RE~~3~~2, will match a compressed or partially compressed GS1 Digital Link URI, with the same caveats as for RE1.

RE~~2~~3:

```
^https?:(\/\/((([^\/?#]*)@)?([^\/?#:]*)(:([^\/?#]*))?))?([^?#]*)((\/[0-9A-Za-z_-]{10,}$))
```

As a further warning, recall that a GS1 Digital Link URI may contain arbitrary path segments between the domain name and the primary key. It is unlikely but possible that those path segments will all be numeric. An even less likely scenario, but still possible, is that the compression algorithm may create an all-numeric output. RE~~2~~3 must allow for this but the downside is that a URL like https://example.com/0123/456789012340123, which is **not** a valid uncompressed GS1 Digital Link URI and will fail to match RE1~~ or RE2~~, will match RE3.

~~6.1.6~~6.1.3 Recommended procedure

Noting the limitations of regular expressions in the previous sections, we do not recommend combining RE3 with RE1~~ or RE2~~. Rather, the recommended procedure is that for a given input string:

1. If a given string matches ~~RE2 or~~ RE1 (in that order), it plausibly is an uncompressed GS1 Digital Link URI.
2. If it doesn't match that, but does match RE~~2~~3, then it plausibly is a compressed GS1 Digital Link URI.
3. If it matches none of the regular expressions here, it definitely is not a conformant GS1 Digital Link URI, compressed or otherwise.

6.2 Human Readable Interpretation (HRI)

This standard defers entirely to the GS1 General Specifications for rules concerning human-readable interpretation.

7 Glossary

The glossary lists the terms and definitions that are applied in this document. Please refer to the *www.gs1.org/glossary* for the online version.

Term	Definition
Attribute	An element string that provides additional information about an entity identified with a GS1 identification key, such as batch number associated with a Global Trade Item Number (GTIN).
Brand Owner	The organisation that owns the specifications of a trade item, regardless of where and by whom it is manufactured. The brand owner is normally responsible for the management of the Global Trade Item Number (GTIN).
Canonical GS1 Digital Link URI	The definitive GS1 Digital Link URI for a given resource. See section *4.12*
Consumer	Often considered as the "recipient" of the supply chain in the past, today's consumer is an active part of the supply chain and expects more data, with higher accuracy, and greater ease.
Consumer Product Variant (CPV)	An alphanumeric attribute of a GTIN assigned to a retail consumer trade item variant for its lifetime.
Data Field	A field that contains a GS1 identification key, an RCN, or attribute information
Data titles	Data titles are the abbreviated descriptions of element strings which are used to support manual interpretation of barcodes.
Dereferencing a URI	The use of an appropriate access mechanism (e.g. Web request) to perform an action on the URI's resource (e.g. to retrieve an information representation via HTTP GET or to send data to a resource via an HTTP POST operation). Dereferencing a URI is often considered synonymous with making a Web request or 'looking up' a URI on the Web.
Domain name	A domain name is an identification string that defines a realm of administrative autonomy, authority or control within the Internet. Domain names are formed by the rules and procedures of the Domain Name System (DNS). Any name registered in the DNS is a domain name. Domain names are used in various networking contexts and application-specific naming and addressing purposes. Domain names provide an abstraction layer that separates a registered name for an organisation or activity from the actual internet addresses (IP addresses) that provide its associated information services such as its Website, its e-mail server etc. The system that connects the domain names with the corresponding IP addresses is the Domain Name System (DNS).
Element string	The combination of a GS1 Application Identifier and GS1 Application Identifier data field.
GS1 Application identifier	The field of two or more digits at the beginning of an element string that uniquely defines its format and meaning.
GS1 Application identifier data field	The data used in a business application defined by one GS1 Application Identifier.
GS1 Barcode	A data carrier which encodes GS1 Application Identifier element strings.
GS1 Barcode using GS1 Application Identifiers	All GS1 endorsed barcode symbologies that can encode more than a GTIN namely GS1-128, GS1 DataMatrix, GS1 DataBar and Composite and GS1 QR Code.
GS1 Identification key	A unique identifier for a class of objects (e.g. a trade item) or an instance of an object (e.g. a logistic unit).
GS1 key qualifier	A key qualifier is an additional attribute that is designated for use as part of a compound key (e.g., GTIN + serial number is a compound key, with the serial number being a key qualifier for the GTIN)
GS1 Digital Link URI	A Web URI conforming to the GS1 Digital Link URI syntax.

Term	Definition
key=value pair	The query string of a URL – the portion after the ? symbol - may contain one or more keys, also known as parameters, and their values. For example, an expiry date (17) can be given a value of 221225 as 17=221225. Multiple key=vale pairs can be included in the query string, separated by the & character.
LGTIN (GTIN + Lot/Batch)	A compound key formed from the combination of GTIN [AI (01)] and Batch/Lot identifier [AI (10)]. LGTIN is defined as an EPC Class URN in the current GS1 Tag Data Standard (v1.11), sections 6.4.1 and 7.14, which describes the mapping between the EPC Class URN format for LGTIN and the corresponding element string.
Parsing	The process of analysing the structure of a sentence or URI structure in order to extract relevant information from it. Note that within the context of EPC URN structures, parsing refers to the ability to extract structural components within the EPC structure, e.g. for the purpose of matching against EPC URN patterns.
QR Code®	A two-dimensional matrix symbology consisting of square modules arranged in a square pattern. The symbology is characterised by a unique finder pattern located at three corners of the symbol. QR Code® symbols are read by two-dimensional imaging scanners or vision systems
Reference GS1 Digital Link URI	A GS1 Digital Link URI that uses the id.gs1.org domain
Resolver	The term 'resolver' is not unique to GS1. It is the name for any service that accepts an identifier as input and passes the request about the identified item to information about it. In the GS1 context, a resolver connects a GS1-identified item to one or more online resources that are directly related to it. The item may be identified at any level of granularity, and the resources may be either human or machine readable. Examples include product information pages, instruction manuals, patient leaflets and clinical data, product data, service APIs, marketing experiences and more. GS1 resolvers are defined in [DL-Resolution]
Retailer	An organisation engaged in the sale and distribution of products to consumers. Also includes online retailers/e-tailers
SGTIN (Serialised GTIN)	A compound key formed from the combination of a GTIN [AI (01)] with Serial Number [AI (21)] which provides globally unique identification for every instance of a product. The term SGTIN appears in section 6.3.1 and 7.1 of the current GS1 Tag Data Standard, v1.11
Subdomain	A subdomain is a domain that is part of a main domain. Although example.com is a subdomain of the top-level domain (.com), we most often think of a subdomain as the part of the hostname that precedes the registered domain name. For example, the registered domain name gs1.org has one subdomain ('www') [as in www.gs1.org] that is used for its Website. It also has a subdomain ('id') [as in id.gs1.org] that is used for Web-based data services for GS1.
URI	Uniform Resource Identifier. A string of characters used to identify a resource. The resource may be an information resource such as a Web page or a thing in the real world, such as a physical object, person or location. URIs refer to the superset of Uniform Resource Names (URNs), Uniform Resource Locators (URLs) and Web URIs (which can function both as globally unambiguous names, while also behaving like URLs by enabling intuitive retrieval of related information via the Web).
URI fragment identifier	The fragment identifier component of a URI allows indirect identification of a secondary resource by reference to a primary resource and additional identifying information. The identified secondary resource may be some portion or subset of the primary resource, some view on representations of the primary resource, or some other resource defined or described by those representations. A fragment identifier component is indicated by the presence of an octothorpe/hash/number sign ("#") character and terminated by the end of the URI. A typical use of a URI fragment identifier is to provide a direct link to a specific section within a very long Web document such as https://www.w3.org/TR/dwbp/#DataIdentifiers

Term	Definition
URI path information	A path consists of a sequence of path segments separated by a slash ("/") character. A path is always defined for a URI, though the defined path may be empty (zero length). The path component contains data, usually organized in hierarchical form, that, along with data in the non-hierarchical query component, serves to identify a resource within the scope of the URI's scheme and naming authority (if any). The path is terminated by the first question mark ("?") or number sign ("#") character, or by the end of the URI.
URI query string	The query component contains non-hierarchical data that, along with data in the path component, serves to identify a resource within the scope of the URI's scheme and naming authority (if any). The query component is indicated by the first question mark ("?") character and terminated by a number sign ("#") character or by the end of the URI.
URL	Uniform Resource Locator (URL), a specific type of URI colloquially known as Web address. A URL is a URI starting with http or https .

8 Changes since version 1.1

The single GS1 Digital Link standard version 1.1 has been split into four separate documents:

- GS1 Digital Link Standard: URI syntax (this document)
- GS1 Digital Link Standard: Resolution
- GS1 Digital Link Standard: Compression and decompression
- GS1 Digital Link Standard: Semantics

The canonical version of a GS1 Digital Link URI, section 4.12, is now defined as using HTTPS and the key=value pairs in the query string are now sorted in lexical order of the AIs.

AIs in the 410-416 range, except 414 and 417, have been removed from the list of primary keys (section 4.3).

The ABNF grammar has been updated to support new AIs in the range 4300-4326 introduced in the GS1 General Specifications as a result of GS1 Scan4Transport.

AIs 3950 – 3953 added

ABNF for extension parameters made more precise to match normative text

All primary keys can be included in the query string to accommodate situations where a single URI needs to carry more than one primary key (section 4.10).

Convenience alphas deprecation notice (4.1, 0)

New paragraph emphasising that GS1 DL URIs do not have to, and SHALL NOT be assumed to, point to a resolver (section 2).

Updated introduction.

Example GS1 identifiers changed to use the 952 prefix.

AIDC Considerations section added, including regular expressions for determining strings that definitely are not, or plausibly are, GS1 Digital Link URIs.

8.1 Changes since version 1.2

A number of minor inaccuracies were detected in how the URI syntax reflected the definition of GS1 identifiers. The purpose of the update to this version (1.2.1) was simply to correct these errors. The identifiers affected are:

- The final component of GCN, GDTI and GRAI are all optional (previous versions of the GS1 Digital Link URI Syntax suggested at least one serialisation character was required).

- The final component of GCN is all optional (previous versions of the GS1 Digital Link URI Syntax suggested at least one serialisation character was required).
- GMN now X..25 in line with changes in the GenSpecs between versions 19 and 21.
- Harvest date (7007) now accepts 6 or 12 digits but not intermediate numbers of digits.
- New AI 715, National Healthcare Reimbursement Number – USA NDC, added
- AIs 90, 240, 241, 242, 243, 250, 251, 400, 403, 8102 corrected all updated to allow CHAR, previously they were shown in GS1 Digital Link as being DIGIT only.
- AI 723s now 2*XCHAR 1*28XCHAR (previously it had been written as 2*30XCHAR)
- AI 8008 changed from 8DIGIT 1*4DIGIT to 8DIGIT [2DIGIT] [2DIGIT] which then tolerates 8 digits, 10 digits (if minutes are specified) or 12 digits (if minutes and seconds are specified) but does not tolerate 9 digits or 11 digits.
- It was recognised that `locNoParameter` is redundant because we also have `glnParameter` that expresses exactly the same thing.

8.2 Changes since version 1.2.1

- AI 415, Pay To, reinstated as a primary key (see, in particular, section 4.3)
- Convenience alphas, deprecated in version 1.2.0, have now been removed so that, for example, 'gtin' cannot be used instead of '01' in the path. This change is reflected throughout the document, starting with section 4.3.

9 References

[DL1]

GS1 Digital link version 1.0 Originally titled GS1 Web URI Structure. Mark Harrison, Phil Archer, Dominique Guinard et al. GS1 Ratified Standard, August 2018 *https://www.gs1.org/standards/Digital-Link/1-0*

[DL 1.1]

GS1 Digital Link version 1.1 Mark Harrison, Phil Archer, Dominique Guinard et al. GS1 Ratified Standard, February 2020 *https://www.gs1.org/docs/Digital-Link/GS1_Digital_link_Standard_i1.1.pdf*

[DL-Compression]
GS1 Digital Link: Compression and decompression. Mark Harrison, GS1 ratified standard, see *https://www.gs1.org/standards/gs1-digital-link*

[DL-Resolution]
GS1 Digital Link: Resolution. Phil Archer, Mark Harrison, Dominique Guinard et al. GS1 ratified standard, see *https://www.gs1.org/standards/gs1-digital-link*

[DL-Semantics]
GS1 Digital Link: Semantics, Mark Harrison, Phil Archer et al.GS1 ratified standard, see *https://www.gs1.org/standards/gs1-digital-link*

[GENSPECS]
GS1 General Specifications V22.0. GS1 Ratified Standard January 2022
https://www.gs1.org/sites/default/files/docs/barcodes/GS1_General_Specifications.pdf

[GS1 Identification Keys]
https://www.gs1.org/standards/id-keys

[GS1Voc]
The GS1 Web vocabulary *https://www.gs1.org/voc/*

[IRIs]
Internationalized Resource Identifiers (IRIs) M Duerst, M. Suignard. IETF January 2005
https://tools.ietf.org/html/rfc3987

[Linked Data]
Tim Berners-Lee 2006 *https://www.w3.org/DesignIssues/LinkedData*

[LMS]
GS1 Lightweight Messaging Standard for Verification of Product Identifiers, Release 1.1. GS1 ratified standard July 2019
https://www.gs1.org/sites/default/files/docs/standards/gs1_lightweight_verification_messaging_standard_v1-1.pdf

[PercentEncoding]

Uniform Resource Identifier (URI): Generic Syntax, section 2.1: Percent-Encoding T Berners-Lee, R Fielding, L Masinter. IETF January 2005 https://tools.ietf.org/html/rfc3986#section-2.1

[REST]

See https://en.wikipedia.org/wiki/Representational_state_transfer

[RFC 2606]

Reserved Top Level Domain Names D Eastlake, A Panitz. IETF June 1999 https://tools.ietf.org/html/rfc2606

[RFC 3986]

Uniform Resource Identifier (URI): Generic Syntax. T Berners-Lee, R Fielding, L Masinter. IETF January 2005 https://tools.ietf.org/html/rfc3986

[RFC 5234]

Augmented BNF for Syntax Specifications: ABNF. D Crocker (ed), P Overell. IETF January 2008 https://tools.ietf.org/html/rfc5234

[RFC 6570]

URI Template. J. Gregorio, R. Fielding, M. Hadley, M. Nottingham, D. Orchard. IETF March 2012 https://tools.ietf.org/html/rfc6570

[RFC 6596]

The Canonical Link Relation. M Ohye, J Kupke. IETF April 2012 https://tools.ietf.org/html/rfc6596

[RFC 6761]

Special-Use Domain Names. S Cheshire, M Krochmal. IETF February 2013 https://tools.ietf.org/html/rfc6761

[RFC 7405]

Case-Sensitive String Support in ABNF. P. Kyzivat. IETF December 2014 https://tools.ietf.org/html/rfc7405

[RFC 8615]

Well-Known Uniform Resource Identifiers (URIs). M Nottingham. IETF May 2019 https://tools.ietf.org/html/rfc8615