



### The Latest News— Unlocked and Straight to Your Inbox.

Thanks for reading. Subscribe to get  
the latest blogs, podcasts and  
notifications.

**SUBSCRIBE**

## Data Vault 2.0 Suggested Object Naming Conventions

best practices, data vault standards, dv standards, dv2 standards, naming conventions, standards



A prefix or suffix pattern must be chosen for each object type: processes, views, procedures, functions, etc.

- Each object type listed in #1 must apply the chosen pattern globally (everywhere) for consistency and repeatability. If the Steward team decides a prefix is best for views, then ANY view built and released to production for Data Vault (any company) **must** adhere to a prefix notation. The benefits of following this pattern are: faster time to market, easier member on-boarding, easier (and less) documentation required.
- Common (enterprise wide) documentation of the naming conventions must be created and maintained.
- Any object abbreviations chosen **must not conflict** with each other, or with source system object names –

That's it for the standards. Now for the best practices, and suggested naming conventions that we've been using for years.

## Best Practices and Suggested Naming Conventions

The most important thing to remember is: what ever naming conventions you choose – **must** be enforced / enforced. If your team decides to implement aren't worth anything without enforcement in the TEST area. Your QA/QC team must enforce the standards during development when the standards are broken, and NOT let them slip through to production. Consistency, scalability, success of sticking to the standards.

Naming conventions prove not only useful, but essential in handling multiple geographically split teams and ensuring consistency. They also are essential in delivering **accurate and understandable** production builds.

- Processes: P, PRC, PROC, SP
- Views: V, VW
- Functions: F, FUNC, FNC, FUN
- Tables:
  - Stage: STG, S, STAGE
  - Hub: H, HUB, HB
  - Link: L, LNK, LINK
  - Satellite: S, SAT
  - Hub-Sat: HS, HSAT
  - Link-Sat: LS, LSAT
  - Hierarchical Link: HL, HLNK, HLINK
  - Same-As Link: SAL, SALNK, SALINK, SL, SLINK
  - Effectivity Sat: ES, ESAT, EFFSAT, EFF\_SAT
  - Status Sat: STATUS, ST\_SAT, STAT\_SAT, SSAT
  - Non-Historized Link: NHL, NHLNK, NHLINK, NHLINK, XLINK, XL, XLNK

### The Latest News— Unlocked and Straight to Your Inbox.

Thanks for reading. Subscribe to get  
the latest blogs, podcasts and  
notifications.

First Name

Last/SurName

Email



- Hash Diff: HD, HDIFF, HDF
- Load-Date Time Stamp: LDT, LDTS, LOAD\_DTS
- End-Date Time Stamp: EDTS, ETS
- Sequence: SQN
- Record Source: RSRC, RECSRC, REC\_SRC, RS
- Applied Date Time Stamp: APPL\_DTS, ADTS, APDTS, ATS
- Sub Sequence: SUB\_SQN, SSQN, SUBSQN
- Status Indicator: STAT\_IND
- Deleted Date Time Stamp: DEL\_DTS, DDTS, DELTS, DLTS
- Snapshot Date Time Stamp: SNAP\_DTS, SNAP\_TS, SNPTS
- Updated Date Time Stamp (used in BDV, Master Data, and Info Marts): UPTS, UPDTS, UDTs, UTS
- Generic Prefixes and Suffixes for other columns
  - Business Keys: BK, BKEY
  - Dependent Child: DEPC, DC
  - Driving Key: DK, DKEY

### The Latest News— Unlocked and Straight to Your Inbox.

Thanks for reading. Subscribe to get  
the latest blogs, podcasts and  
notifications.

First Name

Last/SurName

Email

## A Word about Classification

Naming conventions are good, but are only the **base level** of object identification and purpose. They carry meaning about what the object is supposed to be used for, and where it's needed. That said, there is another level: classification of objects (ie: what should they mean or be at the conceptual level)? For this, we develop taxonomies and ontologies – along with conceptual models.

From an implementation view (using most standard data modeling tools) we highly encourage the use of the COMMENT fields on the table structures, columns, procedures, functions, and views. We suggest leveraging JSON formatted comments to house extra metadata. An example of this is below:

COMMENT FOR A HUB\_CUSTOMER TABLE: { "entity\_class": "HUB" }

COMMENT FOR A STAGING TABLE:

```
{
  "entity-class" : "STAGE",
  "business-key-groups" : [ { "bk-columns" : "field1, field2" },
```





- Can be leveraged by automated QA/QC processes

### Note for Snowflake Customers

If you are on Snowflake, you should (instead) leverage the TAGGING options for these purposes. However that said your approach you should then follow the JSON – Commenting styles.

### The Latest News— Unlocked and Straight to Your Inbox.

Thanks for reading. Subscribe to get  
the latest blogs, podcasts and  
notifications.

First Name

Last/SurName

Email





SUBSCRIBE

The Latest News—  
Unlocked and Straight  
to Your Inbox.

Thanks for reading. Subscribe to get  
the latest blogs, podcasts and  
notifications.

First Name

Last/SurName

Email

View More

Jun 20

M.I.T. Customer Journey

Read More →

Jun 16

Podcast – Fireside Chat with Dan Linstedt, Bill Inmon and Kent Graziano





## The Latest News— Unlocked and Straight to Your Inbox.

Thanks for reading. Subscribe to get  
the latest blogs, podcasts and  
notifications.

First Name

Last/SurName

Email

May 27

### Data Vault 2.1 DRAFT Data Architecture Specification v2.1.0

Worlds first! Release of Draft: Data Vault 2.1 Data Vault Architecture Specification v2.1.0 - get it for FREE by logging in or re-creating your account.

Read More →

**Data Vaults Done Right, Everywhere, Every Time™**





Contact

26 Prospect Street,  
St Albans, VT, 05478 USA

Contact@DataVaultAlliance.com

Helpful Links

- Home
- About
- Shop
- Legal
- Knowledge Base

Connect With Us.  
Follow Us.  
Stay in Touch!



Copyright @ 2023 DataVaultAlliance Holdings, LLC

The Latest News—  
Unlocked and Straight  
to Your Inbox.

Thanks for reading. Subscribe to get  
the latest blogs, podcasts and  
notifications.

First Name

Last/SurName

Email

