

Versionen

- [V1 - V7](#)

V1 - V7

UUIDs (Universally Unique Identifiers) haben mehrere **Versionen (v1-v7)**, die sich vor allem darin unterscheiden, **wie sie erzeugt werden**. Jede Version hat ihren eigenen Zweck und Kompromisse (z. B. zwischen Zufälligkeit, Nachvollziehbarkeit und Sortierbarkeit).

Hier ist eine verständliche Übersicht:

☐☐ UUID v1 – Zeit + MAC-Adresse

Prinzip: basiert auf Zeitstempel + Geräteerkennung

- Enthält:
 - Zeit (100-ns seit 1582 ☐☐)
 - MAC-Adresse des Geräts
- ☐ Vorteile:
 - Hohe Wahrscheinlichkeit eindeutig
 - Zeitlich sortierbar
- ☐ Nachteile:
 - Datenschutzproblem (MAC-Adresse sichtbar)
 - Kann Rückschlüsse auf Erstellungszeit geben

☐☐ Heute oft gemieden wegen Privacy

☐☐ UUID v2 – DCE Security (selten genutzt)

Prinzip: wie v1, aber mit zusätzlicher Benutzer-/Gruppen-ID

- Erweiterung von v1 (POSIX UID/GID)
- ☐ Nachteile:
 - Kaum verbreitet
 - Nicht standardmäßig unterstützt

☐ Praktisch irrelevant in modernen Systemen

☐ UUID v3 – Name-basiert (MD5)

Prinzip: Hash aus Name + Namespace

- Deterministisch:
 - gleicher Input → gleiche UUID
- Verwendet:
 - MD5 (heute kryptografisch schwach)
- ☐ Vorteil:
 - Reproduzierbar
- ☐ Nachteil:
 - MD5 gilt als unsicher

☐ Wird selten noch neu eingesetzt

☐ UUID v4 – Zufällig

Prinzip: komplett zufällige Bits

- 122 zufällige Bits
- ☐ Vorteile:
 - Sehr einfach
 - Keine Infos über Ursprung
 - Sehr geringe Kollisionswahrscheinlichkeit
- ☐ Nachteile:
 - Nicht sortierbar
 - Keine Struktur

☐ **Am häufigsten verwendet**

☐ UUID v5 – Name-basiert (SHA-1)

Prinzip: wie v3, aber mit SHA-1

- Deterministisch

- Verwendet SHA-1 statt MD5
- ☐ Vorteil:
 - Sicherer als v3
- ☐ Nachteil:
 - SHA-1 ist inzwischen auch veraltet

☐ Wird noch genutzt, aber nicht ideal für neue Systeme

☐ UUID v6 – Zeitbasiert (verbessertes v1)

Prinzip: wie v1, aber besser sortierbar

- Zeitstempel vorne angeordnet
- ☐ Vorteile:
 - **Lexikografisch sortierbar**
 - Besser für Datenbanken (Indexe)
- ☐ Nachteile:
 - Teilweise noch Zeitinformationen enthalten

☐ Moderne Alternative zu v1

☐ UUID v7 – Zeit + Zufall (modern)

Prinzip: Unix-Zeit + Zufallsbits

- Enthält:
 - Unix-Timestamp (Millisekunden)
 - Zufällige Bits
- ☐ Vorteile:
 - Sortierbar nach Zeit
 - Keine MAC-Adresse → besserer Datenschutz
 - Gute Balance zwischen Struktur und Zufälligkeit
- ☐ Nachteile:
 - Relativ neu (Support noch im Aufbau, aber wächst schnell)

☐ Kurzvergleich

Version	Basis	Sortierbar	Deterministisch	Typische Nutzung
v1	Zeit + MAC	☐	☐	Legacy
v2	v1 + UID	☐	☐	Kaum genutzt
v3	MD5 Hash	☐	☐	Reproduzierbar
v4	Zufall	☐	☐	Standard
v5	SHA-1 Hash	☐	☐	Reproduzierbar
v6	Zeit (neu sortiert)	☐	☐	DB-optimiert
v7	Zeit + Zufall	☐	☐	Modern / empfohlen

☐ Empfehlung (Stand heute)

- **v4** → wenn dir Einfachheit reicht
- **v7** → wenn du moderne Systeme baust (DB, Skalierung, Sortierung)