

# Gematik Referenzvalidator

Verbindliche Regeln zur Nutzung vom Referenzvalidator | 28.11.2022

Dr. Alexey Tschudnowsky  
Software Architekt @ gematik GmbH

# Motivation oder „Wie validierst Du die Instanzen“?

- Validierungswerkzeuge verhalten sich unterschiedlich („ist es eine wichtige Warnung oder kann sie weg?“)
- Konfigurationseinstellungen beeinflussen das Ergebnis (z.B. unbekannte Profile verboten)
- Die richtigen FHIR-Packages zur Validierung sind nicht immer einfach zu ermitteln (z.B. externe Value Sets)
- Zusätzlich: Profiltüchtigkeitszeitraum?

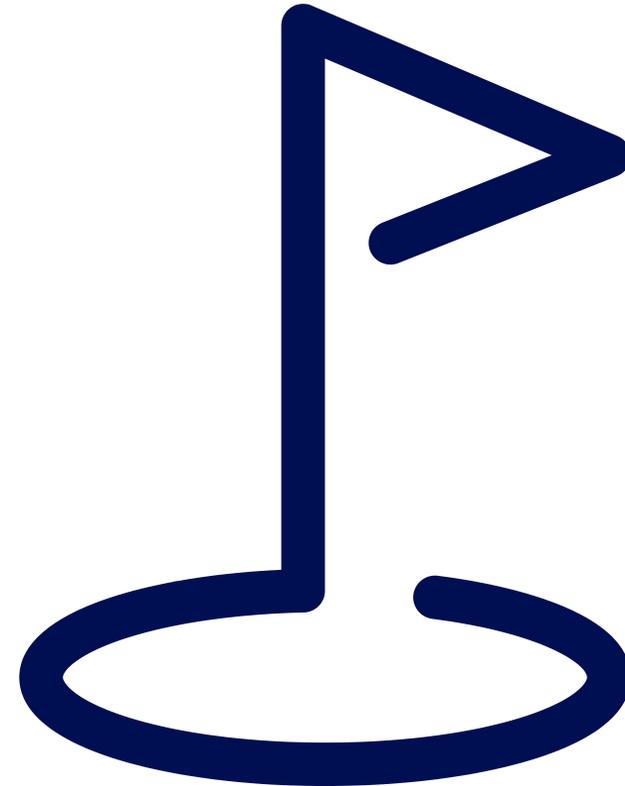


<https://confluence.hl7.org/display/FHIR/Open+Source+Implementations>

Bildquellen: <http://hl7.org/fhir/> <https://fire.ly/>

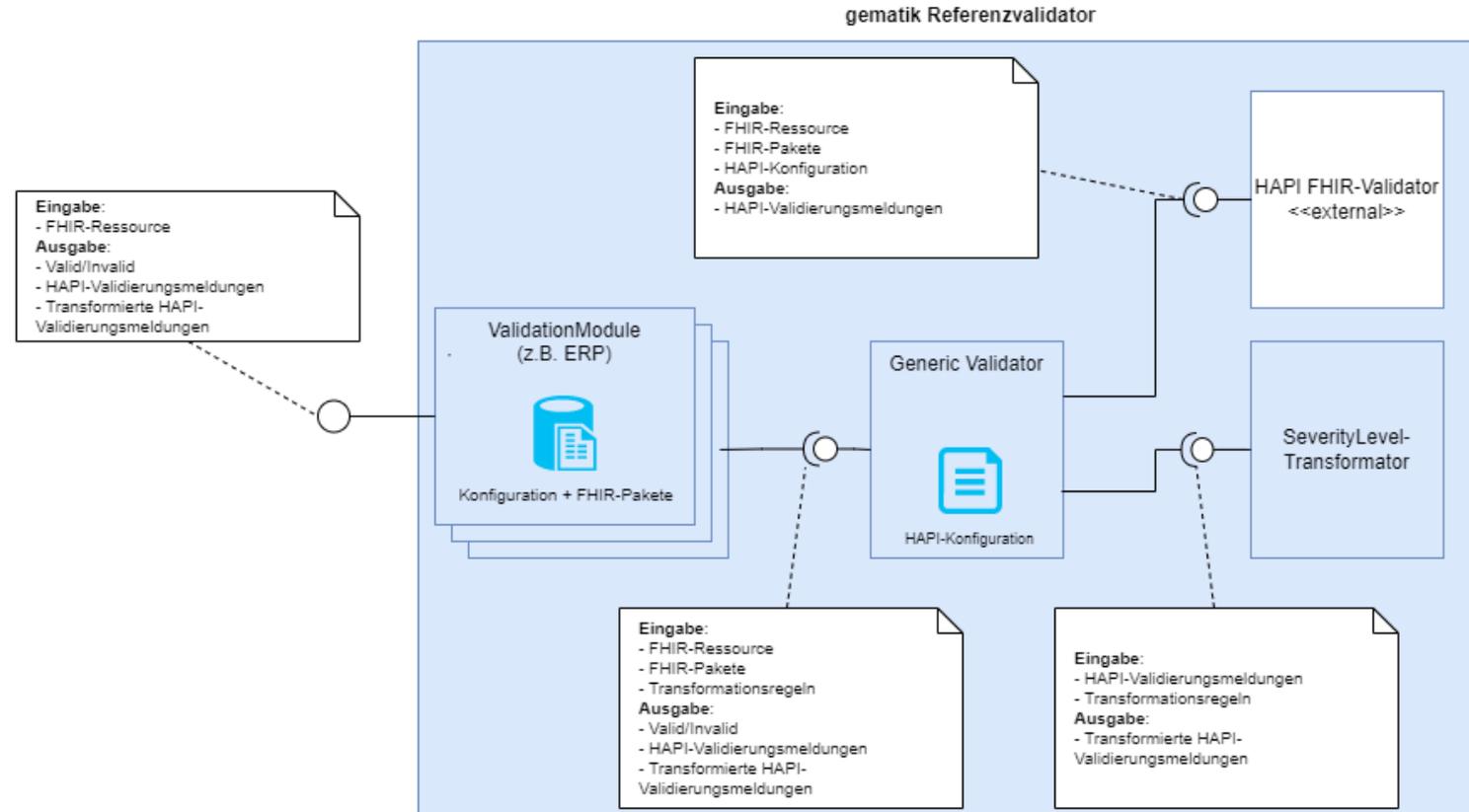
# Ziele

1. **Unterstützung** bei der Umsetzung von (TI-)Spezifikationen
2. **Schiedsrichterfunktion**: Autoritative Aussage in Konfliktfällen (auch für „ältere“ Instanzen)
3. **Referenz** für andere Validatoren
4. Zusätzlich: **Quality Gate** in der TI-Produktivumgebung (TI-Validator)



# Umsetzung

- Prüfgrundlage: **eingebettete FHIR-Profile**
- **Modulare Architektur** zur Unterstützung vieler Anwendungen / Spezifikationen
- Basiert auf dem **HAPI FHIR / HL7 FHIR Validator**
- **Fest verdrahtete Konfiguration** des HAPI / HL7 Validators mit anwendungsspezifischer Behandlung von HAPI Validierungsnachrichten



<https://github.com/gematik/app-referencevalidator/blob/main/docs/concept/concept.md>

# Learnings: Profile und FHIR-Packages

Der Referenzvalidator ist nur so gut wie die zugrundeliegenden Profile

1. Abweichungen von den zugrundeliegenden Informationsmodellen / Spezifikationen werden NICHT berücksichtigt
2. Fehler in den verwendeten FHIR-Packages führen zu „falschen“ Validierungsergebnissen und das solange die Packages unkorrigiert bleiben
3. Mehrere FHIR-Package-Versionen für eine und dieselbe Profilversion sind problematisch



# Learnings: HAPI als Basis

Der Referenzvalidator ist nur so gut wie die darunterliegende Validierung-Engine

1. Implementierung auf HAPI-Basis ist einfach und schnell
2. Hoher Speicherverbrauch und langsamer „Kaltstart“
3. Fehler in der HAPI-Validierung werden „propagiert“



# Aktuelle Herausforderungen

1. Wie „validiert“ man eigentlich ein Prüfmodul?
2. Sind FHIR-Profile als Prüfgrundlage sinnvoll, wenn sie keine autoritative Spezifikation darstellen?
3. Welche Fehler oder Ungenauigkeiten aus den Profilen darf der Validator „heilen“?
4. Kann der Validator als Schiedsrichter seine Validierungsergebnisse trotz Änderungen an HAPI und neuer FHIR-Package-Versionen über längeren Zeitraum stabil halten?
5. Wie validiert man gegen Value Sets, die sich mit der Zeit ändern?



# Vielen Dank für Ihr Interesse!

Dr. Alexey Tschudnowsky  
alexey.tschudnowsky@gematik.de