

**Mit Versionierungsinformationen im  
Requirements Interchange Format (ReqIF)  
echte Wiederverwendung  
von Anforderungen erreichen**

Dr. Oliver Alt

GI-RE-Fachgruppentreffen 2012



- Sitz in Wien
- Softwareentwicklung, Training & Consulting
- Präferierter Partner von Sparx Systems Central Europe
- Model-based Engineering mit Sparx Systems Enterprise Architect
- .net Software Entwicklung
- Multitouch Solutions

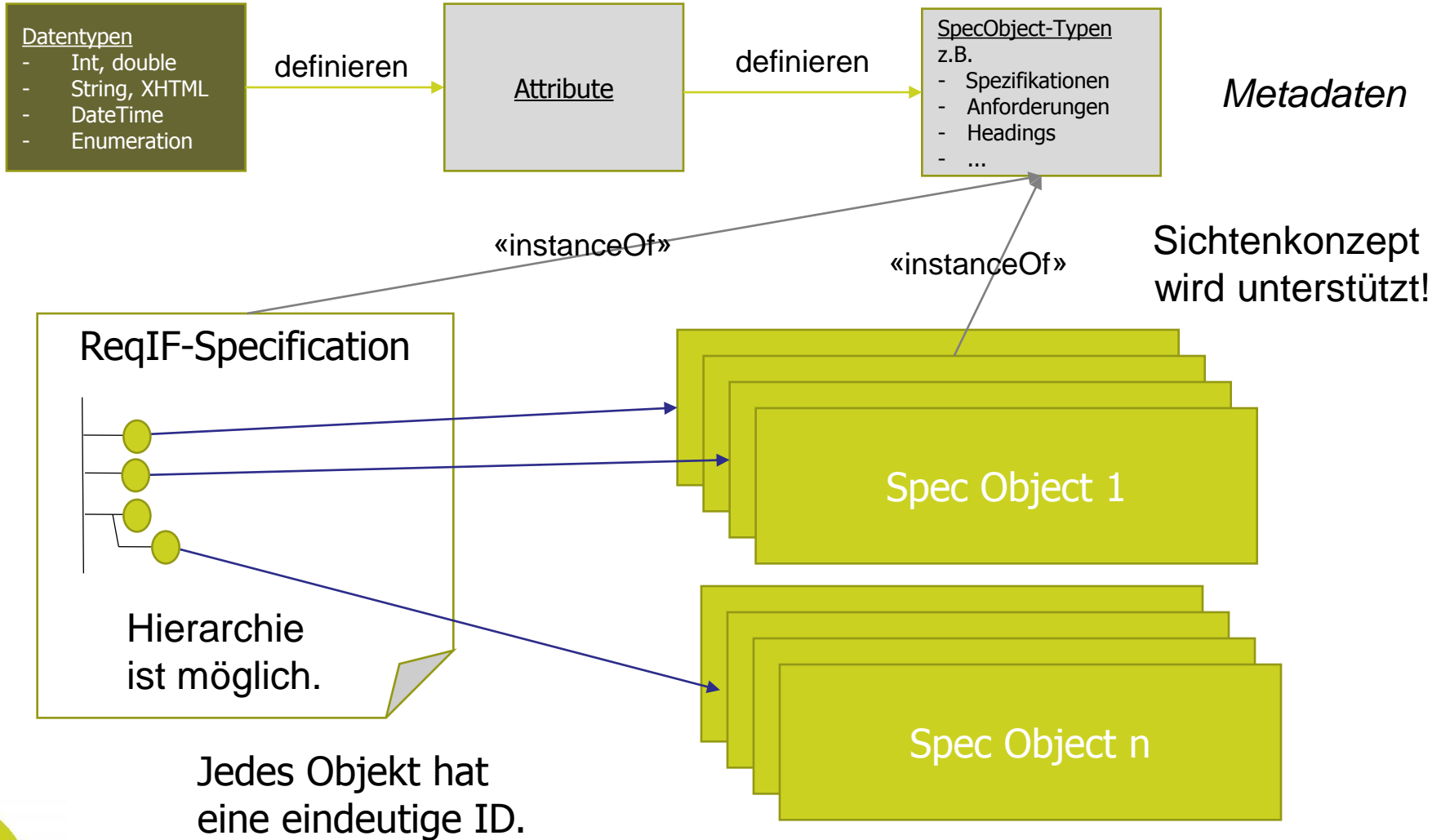
# Motivation

- Bekannte Probleme aus der Praxis
  - Neue Projekte starten oftmals auf Basis eines Vorgängerprojektes
  - Wiederverwendung von Anforderungen oftmals durch Copy & Paste der Inhalte
  - Änderungen sind schlecht nachvollziehbar → Traceability leidet
  - Auch nicht zu verändernde Anforderungen werden kopiert

# ReqIF – Requirements Interchange Format

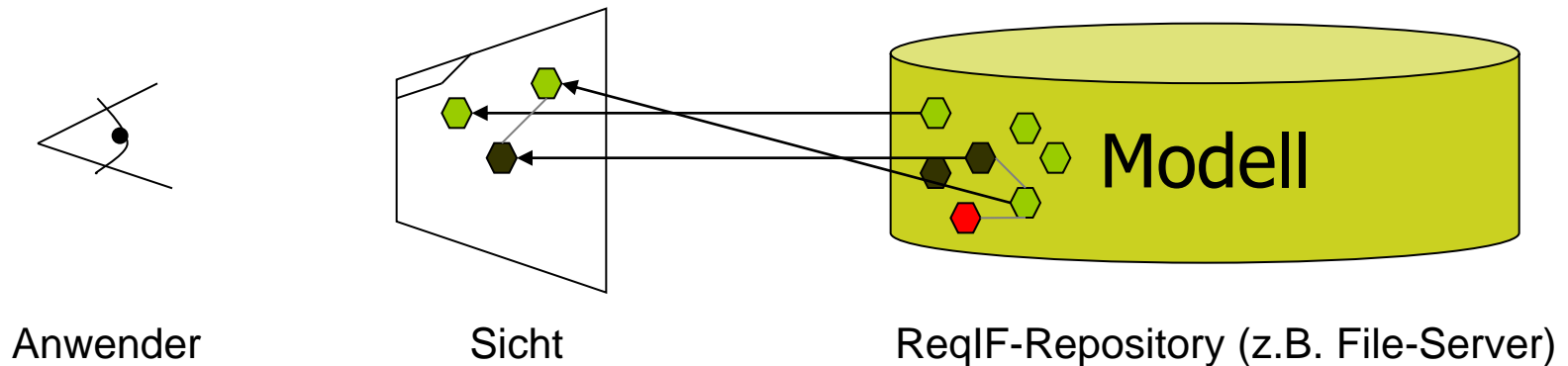
- Standard zum Austausch von Anforderungsspezifikationen
- Hervorgegangen aus RIF der HIS-Arbeitsgruppe
- Seit 2011 OMG Standard
- XML-Format
- Kann aber auch als Datenmodell direkt verwendet werden
- Ähnliche Konzeption wie Metamodellierung

# Konzept von ReqIF



# Modell und Sicht

- *Eine Sicht ist eine Projektion eines Modells, die es von einer bestimmten Perspektive oder einem Standpunkt aus zeigt und Dinge weglässt, die für diese Perspektive nicht relevant sind.*



- In ReqIF bilden die **Spezifikationen** die **Sichten auf das Modell**

# Teilproblem gelöst!

- Die Nutzung des Sichtenkonzepts löst das Problem der Verdoppelung von Daten
- Gleiche Anforderungen können in mehreren Spezifikationen dargestellt werden
- Was aber, wenn sich eine Anforderung ändert?

→ Lösung: Einsatz von Versionierung

# ReqIF mit Versionierung

- Idee: Hinzufügen von Versionsnummern zu den SpecObject-Elementen und Vorhalten der „alten Versionen“ ermöglicht echte Wiederverwendung und volle Nachvollziehbarkeit aller Änderungen

Spec Object mit ID 1/main-1

Spec Object mit ID 1/main-2

Anfügen von Versionsinformationen an die normale ID:

`IDmitVersion := NormaleID + `/` + <branch> + `-` + <version>`



# Versionierung ohne Änderung des ReqIF-Standards

- Durch Erweiterung der ohnehin vorhandenen Ids mit zusätzlichen Versionsinformationen entstehen versionierte Spec-Objekte mit einer neuen ID.
- Auch diese IDs können durch ReqIF-Werkzeuge verarbeitet werden
  - Keine Änderung des Standards nötig
- Zusatzinformationen wie *branchdFrom*, *nextVersion* etc. können durch Attribute ebenfalls unterstützt werden

# Baselining

- Das Sichern von Ständen einer Spezifikation erfolgt durch einfaches Branchen eines Spezifikation mit den Versionen der aktuellen Versionen
- Sofern diese nicht mehr geändert werden – was durch Werkzeuge sichergestellt werden kann – bleibt der Stand immer erhalten
- An der Spezifikation von der gebrannt wurde, kann ganz normal weiter gearbeitet werden

# Problem gelöst!

- Mit Hilfe der Versionierung können alle alten Stände nachvollziehbar gespeichert und wiederhergestellt werden
- Anforderungen können verändert werden, das Verweisen auf alte Stände wird möglich
- Änderungen können zu 100% nachvollzogen werden

# Fazit

- ReqIF ist mehr als nur ein Austauschformat für Anforderungsdaten
- Die dahinterliegenden Konzepte eignen sich auch gut für die Nutzung als direktes Datenformat für Anforderungsmanagementwerkzeuge
- Durch das Sichtenkonzept wird die Wiederverwendung von Anforderungen ohne Datenverdopplung unterstützt
- Die Erweiterung mit Versionierungsinformationen ermöglicht darüber hinaus eine echte Nachvollziehbarkeit aller Änderungen

The screenshot shows a software specification tool interface. At the top left, there is a back arrow icon and the title 'Wetterstation SRS'. Below this is a section titled '1 Introduction'. The first item, '1.1 Wetterstation • main-1', is highlighted in purple and includes the description: 'Die Wetterstation muss die Umgebungstemperatur aufnehmen und auf einem Textdisplay zur Anzeige bringen.' Below it are items 1.2 through 1.6, each with a brief description. At the bottom of the interface, there is a toolbar with five icons: a pencil (Edit Spec Object), a 'V' (Select Version), a downward arrow (New Item Down), an upward arrow (New Item Up), and a group of people (New Child Item). On the right side of the interface, there is a vertical toolbar with various navigation and editing icons.

## Wetterstation SRS

### 1 Introduction

- 1.1 Wetterstation • main-1**  
Die Wetterstation muss die Umgebungstemperatur aufnehmen und auf einem Textdisplay zur Anzeige bringen.
- 1.2 Umgebung • main-1**  
Die Wetterstation muss ihre Daten in einer Umgebung aus Luft erfassen.
- 1.3 Temperatursensorkette • main-1**  
Die Temperatursensorkette muss die Umgebungstemperatur erfassen und als Softwaresignal bereitstellen.
- 1.4 Temperatursensor • main-2**  
Der Temperatursensor muss die Umgebungstemperatur im Bereich  $-40^{\circ}\text{C}$  bis  $+85^{\circ}\text{C}$  erfassen können.
- 1.5 Sensortyp • main-1**  
Der Temperatursensor soll passiv arbeiten.
- 1.6 Applikationssoftware • main-1**  
Die Applikationssoftware soll so entworfen werden, dass spätere Erweiterungen leicht möglich sind.

Edit Spec Object   Select Version   New Item Down   New Item Up   New Child Item

# Vielen Dank!

[www.LieberLieber.com](http://www.LieberLieber.com)

