

Eine Frage vorab...

Wer hier im Saal ist
an der **Entwicklung**
autonomer Fahrzeuge
beteiligt?

Wer kennt die DARPA Grand Challenge?



2004/05
DARPA Grand
Challenge



2007
DARPA Urban
Challenge



„heute“
Googles
autonomes
Fahrzeug

nIrpBench

A Benchmark for Natural Language Requirements Processing

Walter F. Tichy, Mathias Landhäußer, Sven J. Körner

IPD Tichy, Fakultät für Informatik

Wozu Empirie?

- Allgemein: Es gibt unzählige Tools und Verfahren – viel zu viele um sie auszuprobieren.
- Im RE-Bereich
 - macht jeder „etwas anderes“ als die anderen.
 - evaluiert jeder anders als die anderen.
 - sind Evaluationstexte selten verfügbar, nachvollziehbar usw.
- Ein **objektiver Vergleich** ist jedoch kritisch für Praxis *und* Forschung. Ohne Vergleiche vertrauen wir auf Trends, Mode, Meinungen, persönliche Präferenzen, Hörensagen, Consultants, Gurus usw.
- Empirische Studien zeigen auf, **welche Unterschiede** zwischen den Ansätzen **tatsächlich existieren**.

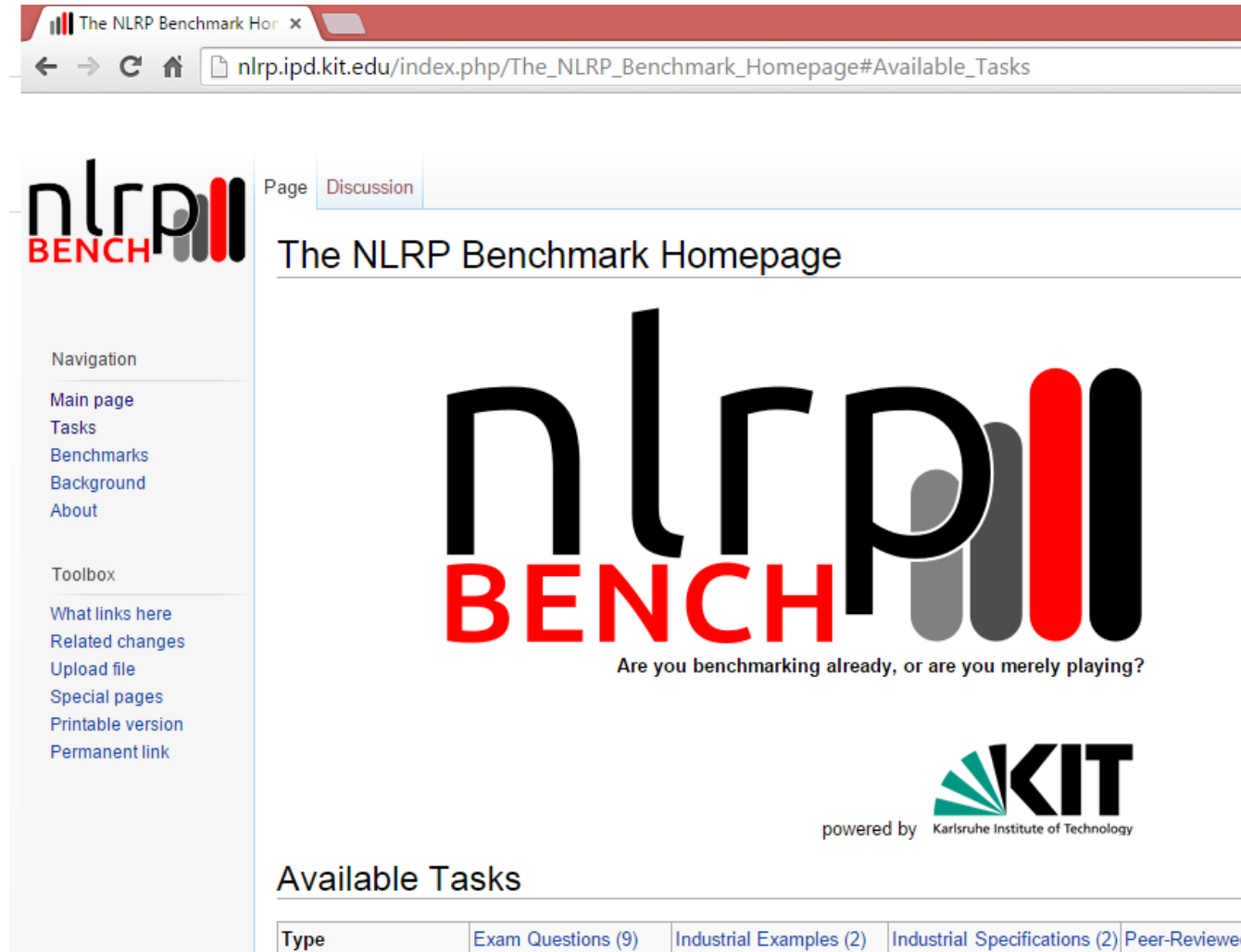
Was ist zu tun?

- Wie kann die Forschungsgemeinschaft den **Stand der Kunst vorantreiben** und das **verlässlich demonstrieren**?
- Und das am besten **schneller als in der Vergangenheit**?
- Kleine Randbemerkung: „Mehr Geld“ ist die falsche Antwort...

Unser Vorschlag: Verwendet Benchmarks!

- Benchmarks sind Sammlungen von **Problemen** mit **Metriken** (und/oder Musterlösungen) zur **Überprüfung der Güte** von eingereichten Lösungen.
- Unabhängige Forscher können ihre Problemlösung an gemeinsamen Problemen testen und mit anderen vergleichen.
- Benchmarks sind nicht kostenlos
 - Probleme müssen gesammelt, aufbereitet und gelöst werden.
 - Metriken und (automatische) Überprüfung von Lösungen müssen vorbereitet werden.
- Vorteile von Benchmarks gegenüber Experimenten in echten Projekten
 - Beinahe kostenfrei wiederholbar. So ist Amortisation möglich!
 - Risikofrei – keine echten Projekte als Versuchskaninchen!
 - Bieten eine Basis für Vergleiche!

nlrpBENCH: Ein Benchmark für die Verarbeitung von Anforderungen



The screenshot shows a web browser window with the URL `nlrp.ipd.kit.edu/index.php/The_NLRP_Benchmark_Homepage#Available_Tasks`. The page title is "The NLRP Benchmark Homepage". The main content features a large logo for "nlrp BENCH" with the text "Are you benchmarking already, or are you merely playing?" below it. The logo consists of the letters "nlrp" in black, "BENCH" in red, and a stylized bar chart to the right. Below the logo is the KIT logo and the text "powered by Karlsruhe Institute of Technology". A sidebar on the left contains navigation links such as "Main page", "Tasks", "Benchmarks", "Background", "About", and a "Toolbox" section. At the bottom, there is a table titled "Available Tasks".

Type	Exam Questions (9)	Industrial Examples (2)	Industrial Specifications (2)	Peer-Reviewed
------	--------------------	-------------------------	-------------------------------	---------------

Die Struktur von nlrpBENCH

- Enthält Anforderungstexte (Tasks)
 - Kategorisiert nach Sprache, Herkunft, ...
 - Verschiedene Aufgaben: Modellierung, Fehlerfindung
 - Musterlösungen, Implementierungen

- Benchmarks
 - Sammlung von Tasks (d.h. Text + Aufgabe)
 - Metriken zur Bewertung
 - Goldstandard
 - Leider: Noch keine automatische Auswertung

- Beispiele für Benchmarks
 - Text → Modell
 - Text-Korrektur

Automatische Modellerzeugung aus Spezifikationen

TEXT → MODELL

Aufgabenbeschreibung

- Gegeben
 - 5 Spezifikationen
 - Unterschiedliche Länge, Domäne, Autoren

- Gesucht
 - Domänen-Modell
 - Als UML-Klassendiagramm

- Metriken [HG2003]
 - Präzision (Wie groß ist der Anteil korrekter Modellelemente?)
 - Ausbeute (Wieviel der Spezifikation wurde modelliert?)
 - Over-Specification (Wieviel richtige Modellelemente gibt es, die nicht im Goldstandard sind?)

Beispiel: Library von Callan [C1994]

Übersicht

- A **library** issues **loan items** to **customers**. Each customer is known as a **member** and is issued a **membership card** that shows a unique member number. Along with the membership number, other details on a customer must be kept such as a **name**, **address**, and **date of birth**. The library is made up of a number of **subject sections**. [...]

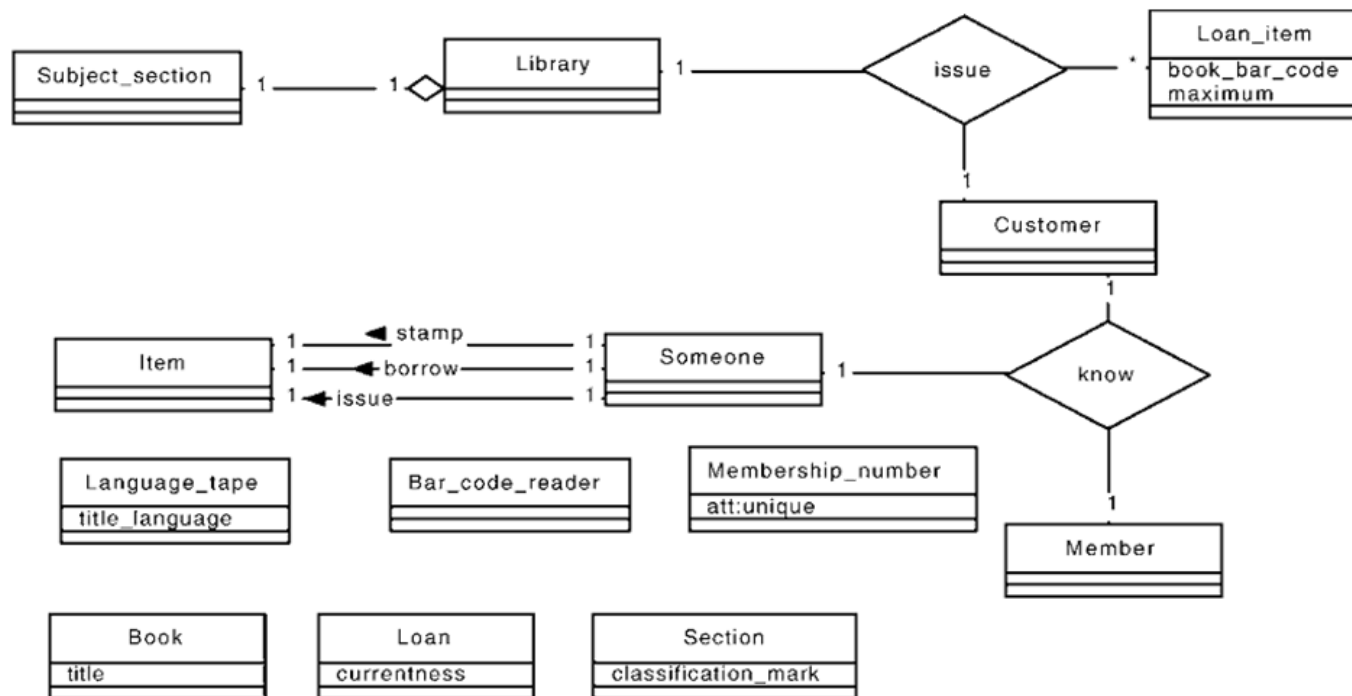


Diagramm aus [HG2003]

Beispiel: Library von Callan [C1994]

Ergebnisse

Lösung	Recall	Precision	F ₁ Measure	Over-Specification
Callan (manuell) [C1994]	30,4%	100%	44,9%	0%
CM-Builder [HG2003]	41,7%	71,4%	52,6%	0%
SAL _E MX [GT2007]	100%	81,4%	89,7%	34,3%

Automatisches Finden von (linguistischen) Fehlern

TEXT-KORREKTUR

Aufgabenbeschreibung

- Gegeben
 - 5 Spezifikationen
 - Unterschiedliche Länge, Domäne, Autoren, ...

- Gesucht
 - Liste von linguistischen Problemstellen bzw. Fehlern

- Metriken
 - Präzision
 - Ausbeute

Beispiel

ABC Video Rental [KZBM2008]

Fehler-Kategorie	#Fehler	Recall Kiyavitskaya [KZMB2008]	Recall RESI [KB2009]
Ähnliche Bedeutungen	15	33%	73%
Unklare Artikel und Quantoren	30	23%	100%
Unvollständigkeiten	23	87%	61%
Nominalisierungen	5	20%	80%

■ Kategorisierung der Fehler

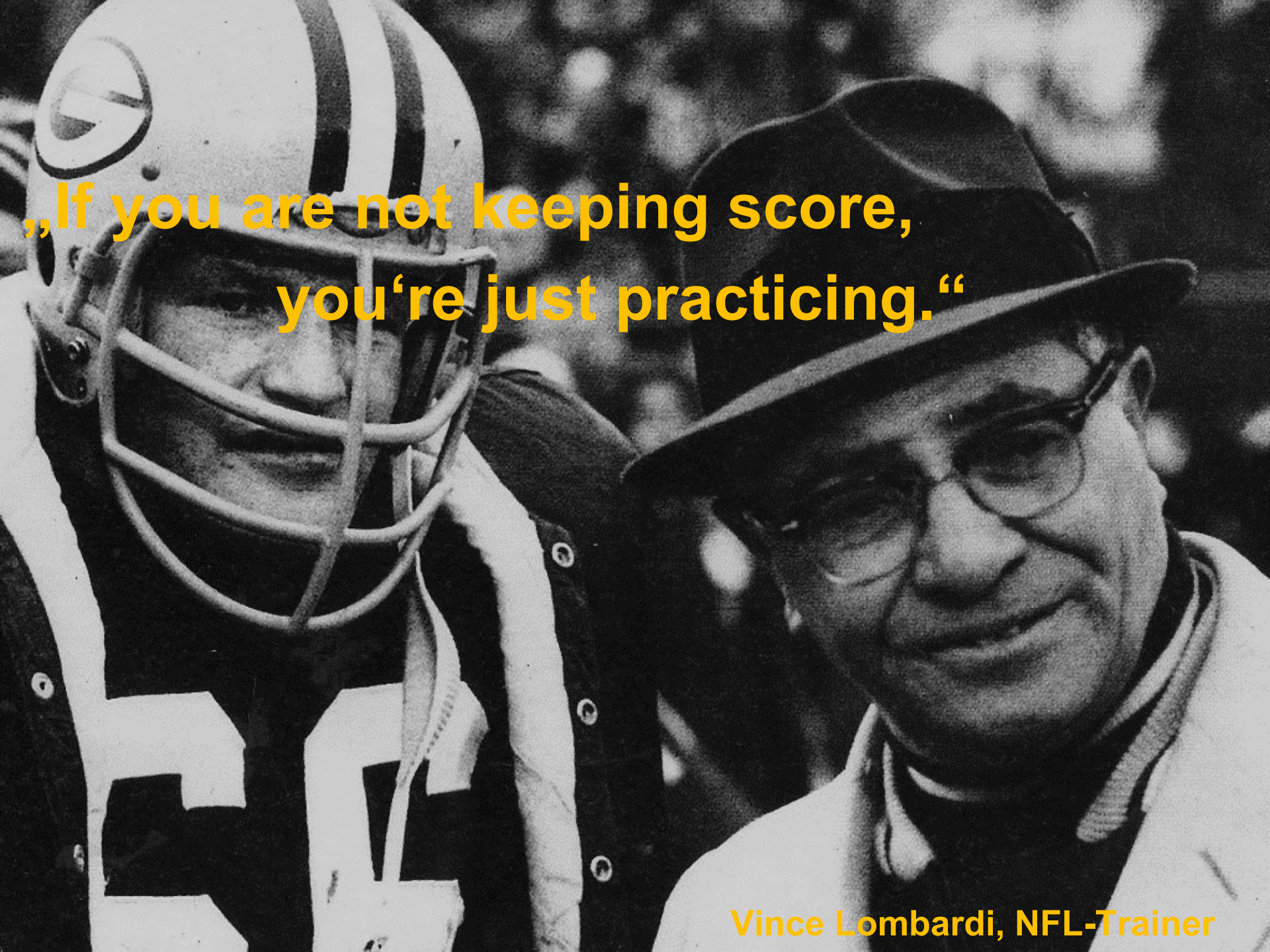
- Ähnliche Bedeutungen: Beinahe-Synonyme verwendet?
- Unklare Artikel und Quantoren: Z.B. „alle X“; wirklich „alle“ auf der Welt?
- Unvollständigkeiten: Z.B. fehlender Agent, fehlendes Objekt (Passiv!)
- Nominalisierungen: Z.B. „Nach der Anmeldung...“: Wer? Wo? Womit?

Schluss

- Benchmarking ist im Software Engineering nicht weit genug verbreitet.
- Mit realistischen Benchmarks erhält man **verlässliche und prüfbare Ergebnisse**.

- Benchmarks
 - schaffen Vergleichbarkeit,
 - eliminieren schlechte Ansätze,
 - helfen, sich auf die Herausforderungen zu konzentrieren,
 - und können in Praxis, Forschung und Lehre eingesetzt werden.

- Die RE-Community
 - sollte Benchmarks **gemeinsam aufbauen** um den Aufwand zu teilen und
 - erst später die richtig teuren Experimente angehen.



„If you are not keeping score,
you're just practicing.“

Vince Lombardi, NFL-Trainer

Literatur

- [C1994] Callan, Robert E. *Building Object-Oriented Systems : An Introduction from Concepts to Implementation in C++*. Southampton: Computational Mechanics Publications, 1994.
- [HG2003] Harmain, H.M., and R. Gaizauskas. “CM-Builder: A Natural Language-Based CASE Tool for Object-Oriented Analysis.” *Automated Software Engineering* 10, no. 2 (2003): 157–81. doi:10.1023/A:1022916028950.
- [GT2003] Gelhausen, Tom, and Walter F. Tichy. “Thematic Role Based Generation of UML Models from Real World Requirements.” In *First IEEE International Conference on Semantic Computing (ICSC 2007)*, 282–89. Irvine, CA, USA: IEEE Computer Society, 2007. doi:10.1109/ICOSC.2007.4338360.
- [KZMB2008] Kiyavitskaya, Nadzeya, Nicola Zeni, Luisa Mich, and Daniel M. Berry,. “Requirements for Tools for Ambiguity Identification and Measurement in Natural Language Requirements Specifications.” *Requir. Eng.* 13, no. 3 (2008): 207–39. doi:10.1007/s00766-008-0063-7.
- [KB2009] Körner, Sven J., and Torben Brumm. “RESI - A Natural Language Specification Improver.” In *Proceedings of the IEEE ICSC 2009*. IEEE, 2009. <http://www.ipd.uni-karlsruhe.de/Tichy/uploads/publikationen/217/ICSC2009.pdf>.