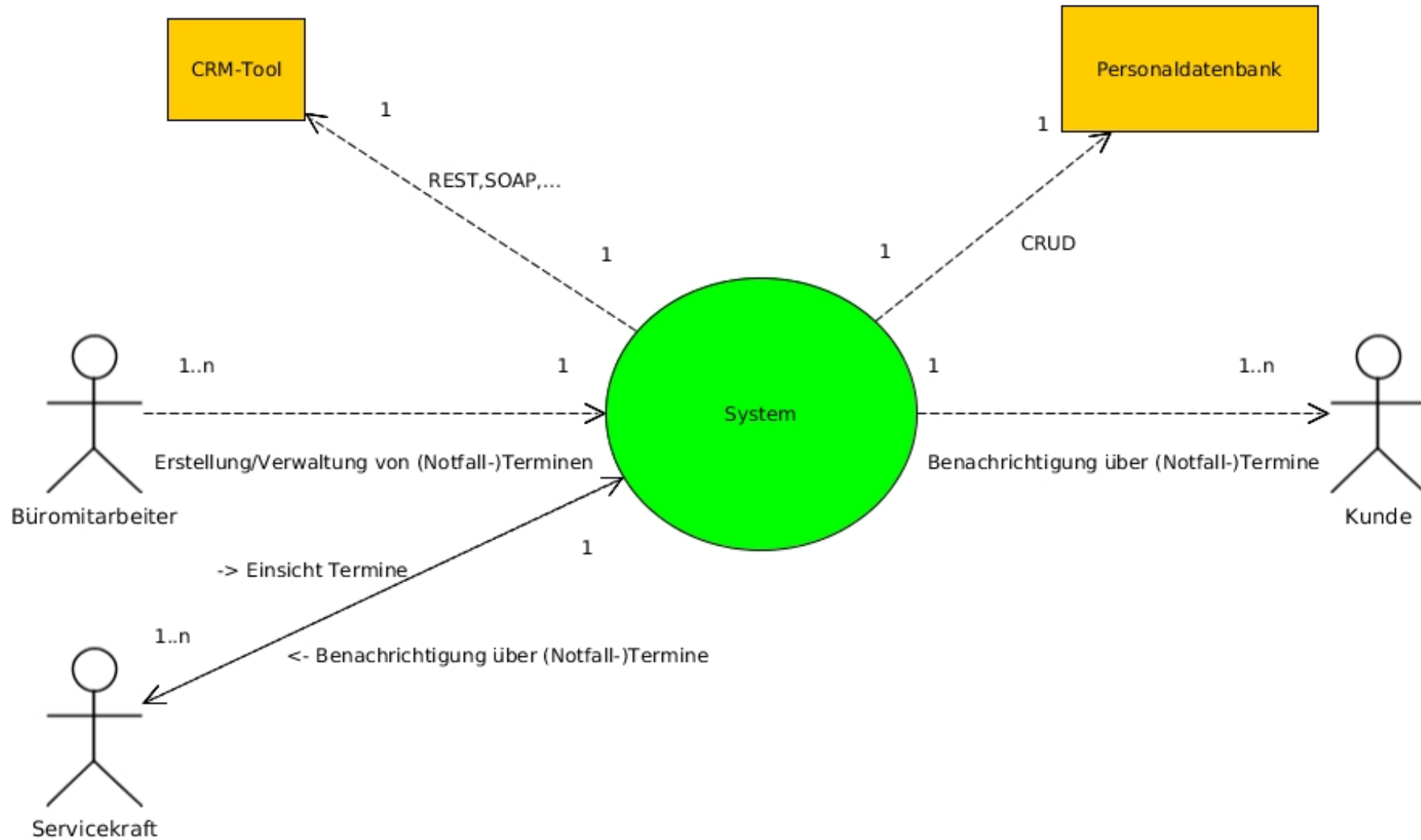


# Zeitplanungssoftware

ACME Corp.

Jan Dieckhoff, Dimitri Meier und Saeed Shanidar

# Kontextdiagramm



# Technologien und Frameworks

- ▶ Frameworks:
  - ▶ Java: Spark, SpringMVC, Play
  - ▶ Python: Flask, Eve
  - ▶ Javascript: express, AngularJS
- ▶ Architektur: REST
  - ▶ Universale Schnittstelle
  - ▶ Leicht umsetzbar
    - ▶ lässt sich leicht als neue Schicht implementieren
  - ▶ Bestehendes System muss nicht großartig verändert werden
  - ▶ Sprachenunabhängig

# Nichtfunktionale Anforderungen

- ▶ Anzahl User
  - ▶ Wieviele Zugriffe pro Tag/Stunde/Minute?
- ▶ Performance
  - ▶ Zeitanforderungen (Echtzeit, Batchorientiert?)
  - ▶ Wieviele Daten werden im Durchschnitt übertragen?
- ▶ Skalierbarkeit
  - ▶ Zentraler Datenverwaltung oder verteilt?
- ▶ Antwortzeiten
  - ▶ Standort des Servers
- ▶ Verfügbarkeit
  - ▶ Wie wichtig ist die Verfügbarkeit, Replikation

# Fragen die mit dem Kunden zu klären sind

- ▶ Zugriff auf die bestehende Personaldatenbank (Standort der Servicekräfte)
- ▶ Automatische Planung von Terminen regelmäßiger Wartungsintervalle für den nächstgelegenen Servicepartner
- ▶ Erstellung von Wochenplänen mit Terminen für jede Servicekraft
- ▶ Erstellung eines Notfalltermins für die nächstgelegene, freie Servicekraft bei Ausfall einer Kundenanlage.