Middleware – Cloud Computing

Koordinierungsdienste (Zusatz)
Nutzung von ZooKeeper zur Anführerwahl

Wintersemester 2022/23

Tobias Distler

Friedrich-Alexander-Universität Erlangen-Nürnberg
Lehrstuhl Informatik 4 (Verteilte Systeme und Betriebssysteme)

Wahl eines Anführers

Konzept

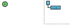
- Problemstellung
  - In einer **Gruppe von Clients** soll ein Anführer gewählt werden
  - Bei Ausfall des Anführers muss ein neuer Anführer bestimmt werden
- Lösungsansatz
  - ZooKeeper-Mechanismus: Dienstunterstützte Knotennamen („Sequenzielle Knoten")
    - Automatisch generierte **Sequenznummer als Teil des Knotennamens**
    - Client spezifiziert beim Anlegen des Knotens den Namenspräfix
  - Erstellen eines „Verzeichnisknotens" `/leader` für die Gruppe
  - Vorgehensweise beim Hinzukommen eines neuen Clients
    - Erstellen eines **flüchtigen Kindknotens** `/leader/node-<Sequenznummer>`
    - Suche nach Kindknoten k_i mit nächstkleinerer Sequenznummer
    - Existiert kein Kindknoten mit kleinerer Sequenznummer → Client ist Anführer
    - Sonst: Client ist kein Anführer → Setzen eines Watch auf den Kindknoten k_i
  - Bei Knotenausfall
    - Automatische Löschung des zugehörigen flüchtigen Knotens
    - **Maximal ein Client** wird per Watch über den Ausfall benachrichtigt

---

Wahl eines Anführers

Beispiel

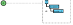
- Start von Client A



Wahl eines Anführers

Beispiel

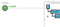
- Start von Client A → Anlegen von node-1



Wahl eines Anführers

Beispiel

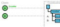
- Start von Client A → Anlegen von `node-1` → **Client A wird Anführer**


Clients
A
Leader
Coordination Service
/
Leader
A:
node-1

Wahl eines Anführers

Beispiel

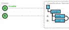
- Start von Client A → Anlegen von `node-1` → **Client A wird Anführer**
- Start von Client B → Anlegen von `node-2`


Clients
A
Leader
B
Coordination Service
/
Leader
A:
node-1
B:
node-2

Wahl eines Anführers

Beispiel

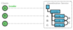
- Start von Client A → Anlegen von `node-1` → **Client A wird Anführer**
- Start von Client B → Anlegen von `node-2` → Watch auf `node-1`


Clients
A
Leader
B
Coordination Service
/
Leader
A:
node-1
B:
node-2
B

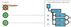
Wahl eines Anführers

Beispiel

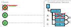
- Start von Client A → Anlegen von `node-1` → **Client A wird Anführer**
- Start von Client B → Anlegen von `node-2` → Watch auf `node-1`
- Start von Client C → Anlegen von `node-3` → Watch auf `node-2`


Clients
A
Leader
B
C
Coordination Service
/
Leader
A:
node-1
B:
node-2
C:
node-3
B
C

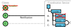
- Ausfall von Client A


Clients
~~Leader~~
B
C
Coordination Service
/
leader
A:
B:
C:
node-1
node-2
node-3
B
C

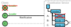
- Ausfall von Client A → Entfernen von node-1


Clients
~~Leader~~
B
C
Coordination Service
/
leader
A:
B:
C:
node-1
node-2
node-3
B
C

- Ausfall von Client A → Entfernen von node-1
- Watch von Client B löst aus

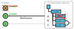

Clients
~~Leader~~
B
Notification
C
Coordination Service
/
leader
A:
B:
C:
node-1
node-2
node-3
C

- Ausfall von Client A → Entfernen von node-1
- Watch von Client B löst aus → **Client B wird Anführer**


Clients
~~Leader~~
B
Leader
Notification
C
Coordination Service
/
leader
A:
B:
C:
node-1
node-2
node-3
C

Wahl eines Anführers

- Ausfall von Client A → Entfernen von node-1
- Watch von Client B löst aus → **Client B wird Anführer**
- Kein Mehraufwand für Client C


Clients
~~Leader~~
B
Leader
C
Notification
Coordination Service
/
leader
A:
B:
C:
node-1
node-2
node-3
C