

6.07.2000

Hauptseminar
**„Methoden zur Spezifikation verteilter
Systeme“**

Vortrag „Statecharts“

Boris Fichgandler

Gliederung

1. Allgemeines
2. Einführung in Statecharts
3. Modellierung sequentieller und paralleler Abläufe
4. NetBill-Protokoll
5. Zusammenfassung

Statecharts im Überblick

- David Harel, The Weizman Institute of Science, Israel, 1984–1986
- Moderne Werkzeuge für die Modellierung mit Statecharts (UML-Standard, Together Enterprise, I-Logix)
- Einsatz von Statecharts in Industrie (BMW, Ford, British Aerospace)

Mealy-Maschinen

- Zustände und Zustandsmengen
- Zustandsübergänge(Transitionen)
- Ereignisse und Aktionen
- Grafische Notation

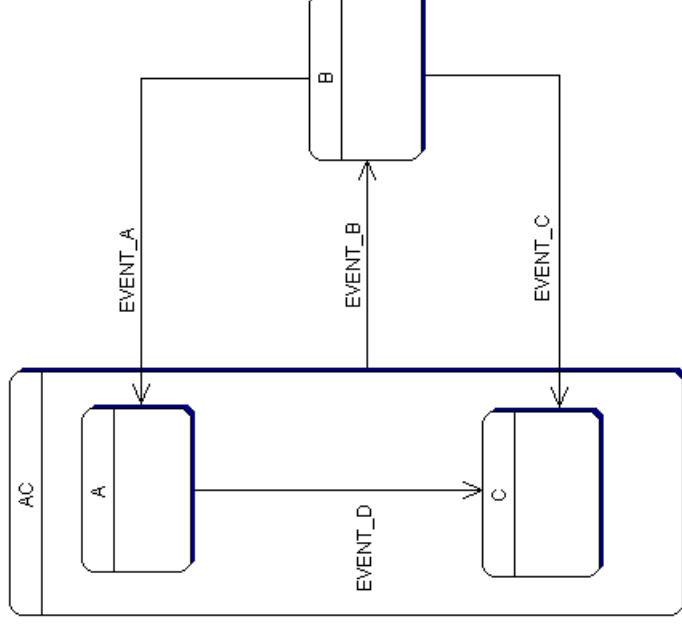
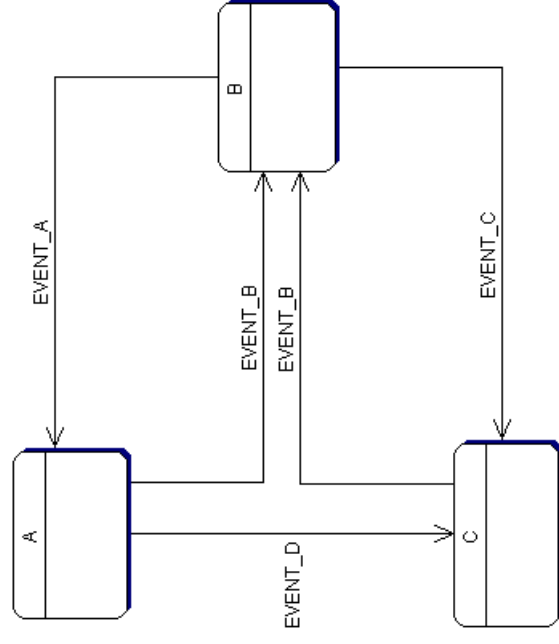


Warum Statecharts?

- Grafische Beschreibung verteilter Systeme mit großen Zustandsräumen
- Clustering und Verfeinerung von Zuständen
- AND/OR-Dekomposition
- Kommunikation und Nachrichtenaustausch
- Verhalten in jeder hierarchischen Schicht ist beobachtbar

Clustering und Verfeinerung

- Zusammenfassung von Zuständen und Zustandsübergängen
- Darstellung eines Zustandes als eine Zustandsmaschine

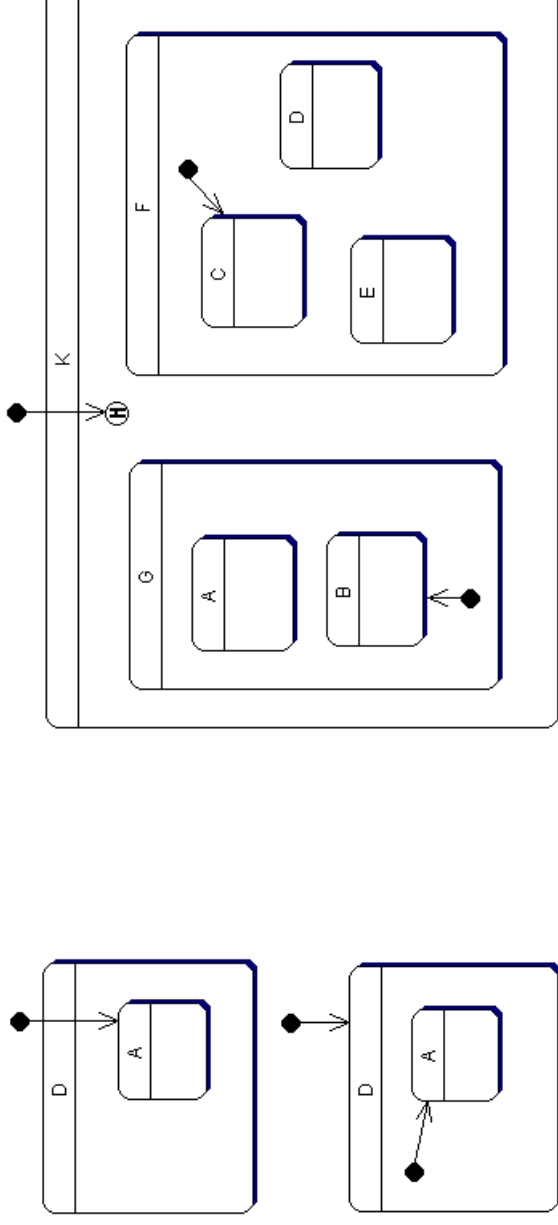


Clustering und Verfeinerung

- Freie Auswahl von Vorgehensstrategien (Top-Down, Bottom-Up)
- Inkrementelles Vorgehen
- „Black Box“-Sicht
- Kompakte Darstellung von Zustandsübergängen (Transitionen)

History-Connector

- Default States
- History-Connector, MRV-Prinzip (Most Recently Visited)
- Erweiterter History-Connector (Deep History)



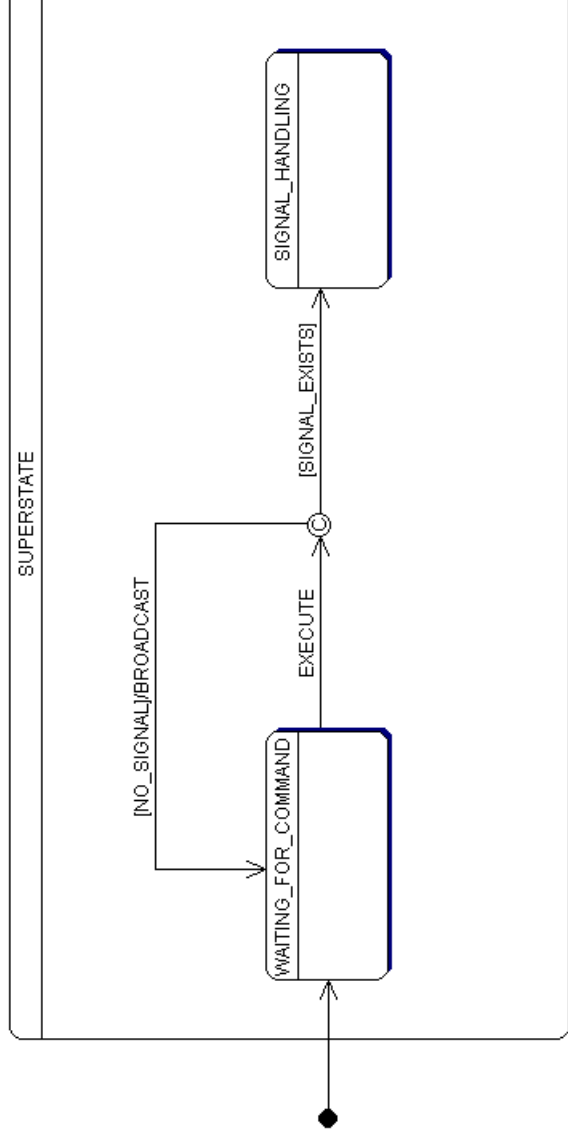
OR-Dekomposition

- XOR-Beziehung zwischen „verbundenen“ Zuständen
- Modellierung sequentieller Abläufe

3. Modellierung sequentieller und paralleler Abläufe

Condition-Connector

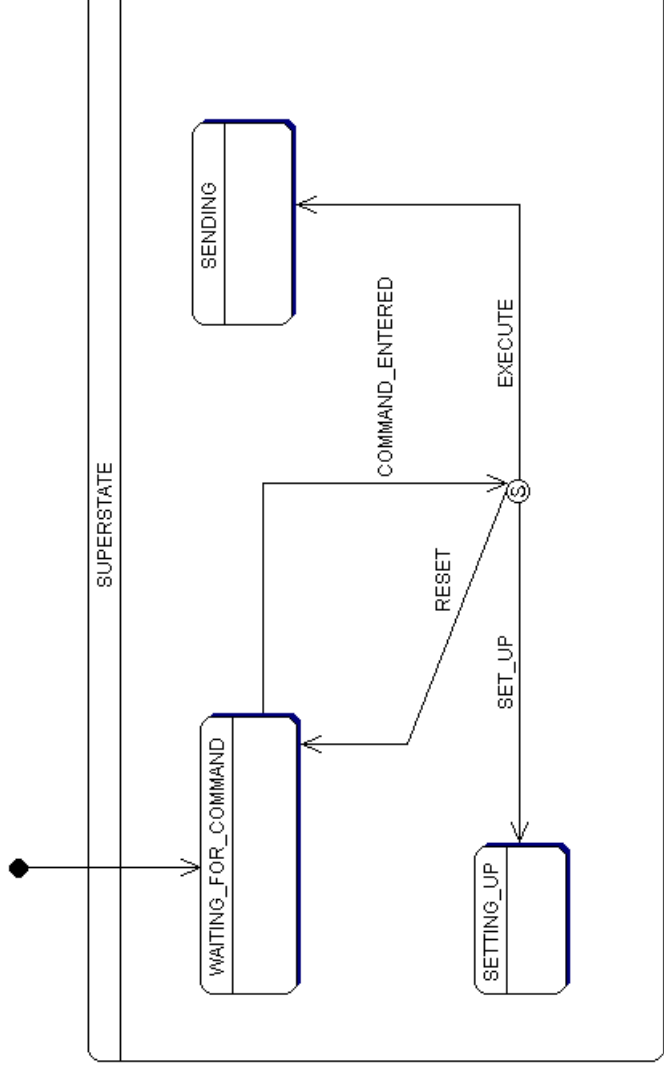
- Zustandsübergang erfolgt bezüglich erfüllter Bedingung



3. Modellierung sequentieller und paralleler Abläufe

Selection-Connector

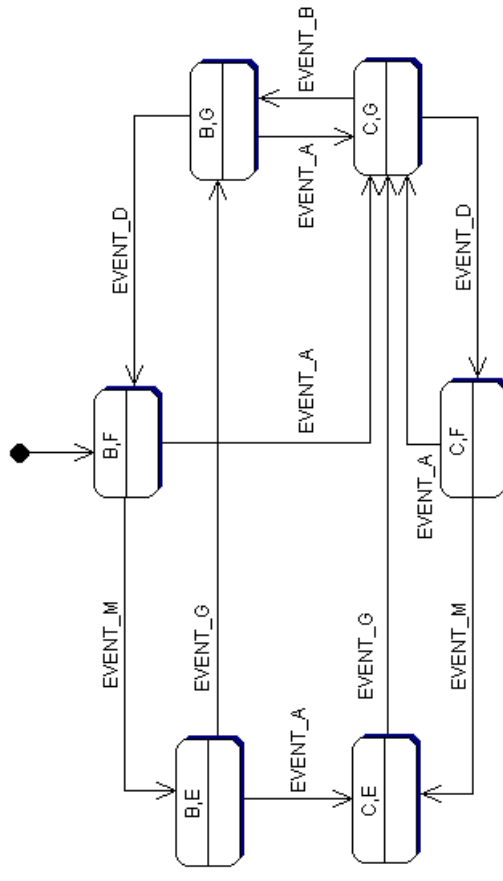
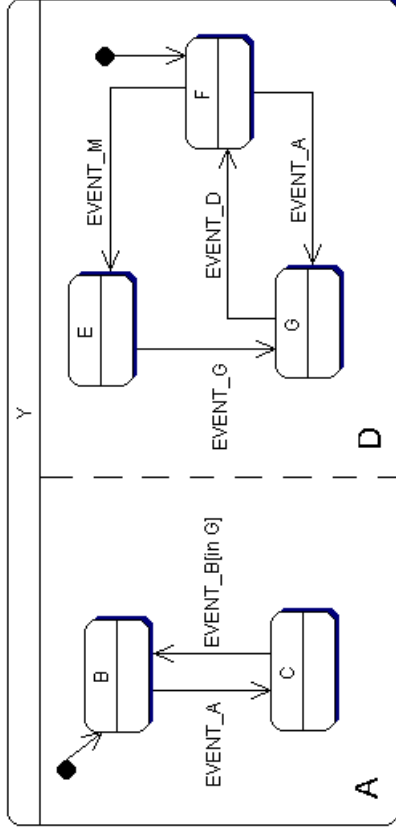
- Unterscheidet zwischen eingetretenen Ereignissen



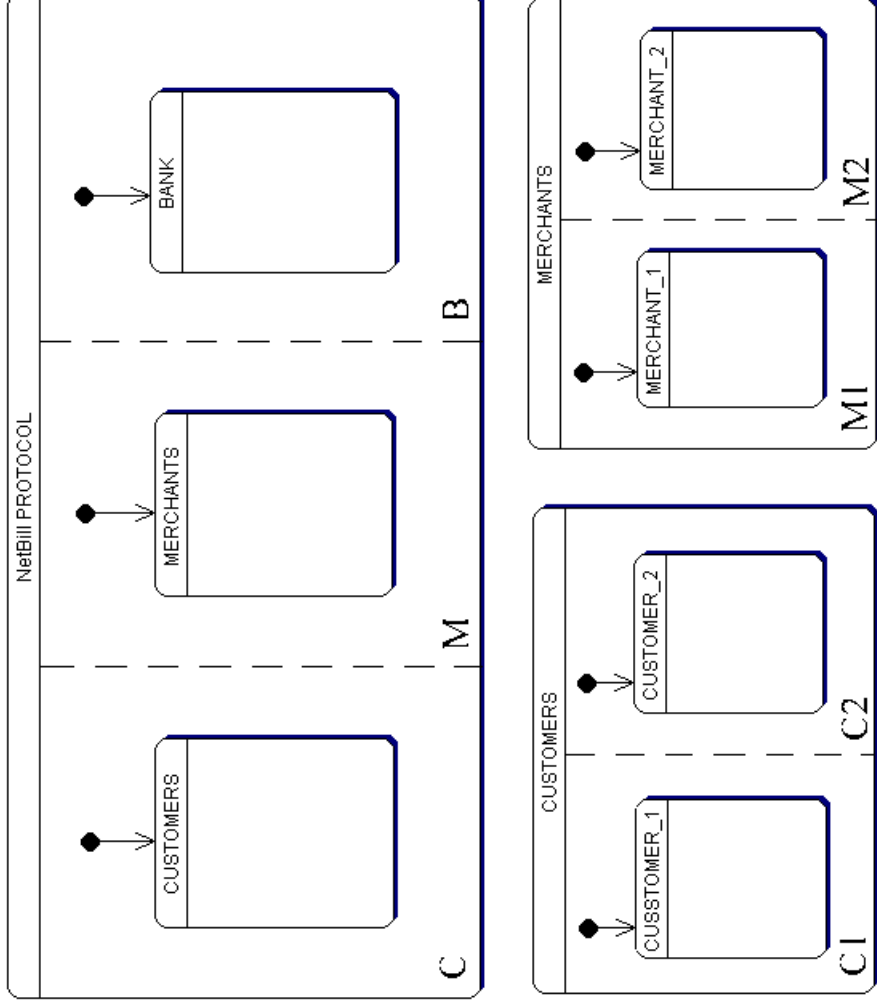
3. Modellierung sequentieller und paralleler Abläufe

AND-Dekomposition

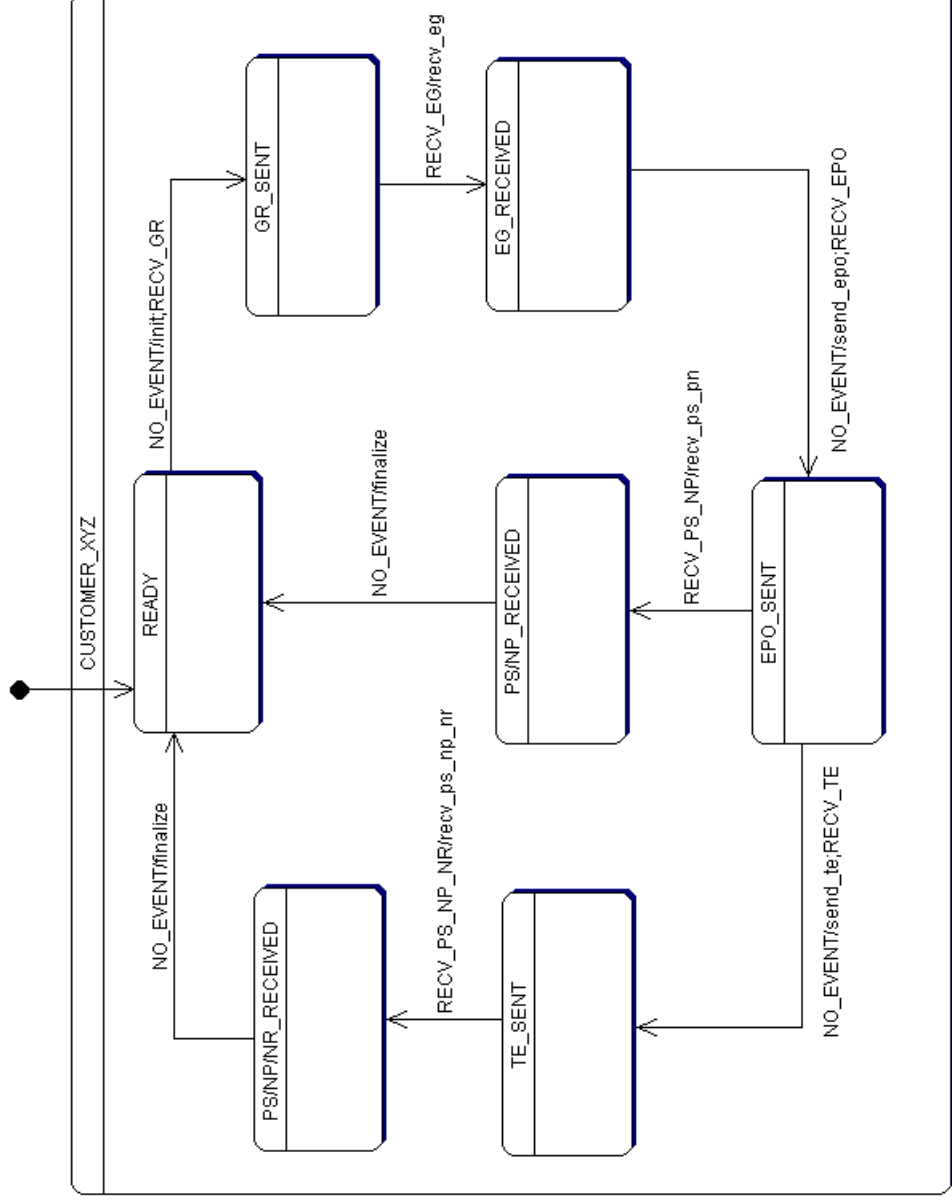
- Modellierung paralleler Abläufe
- Orthogonalität (orthogonales Produkt von Zuständen)
- Synchronisation und Unabhängigkeit
- Kompakte Darstellung von Komponenten



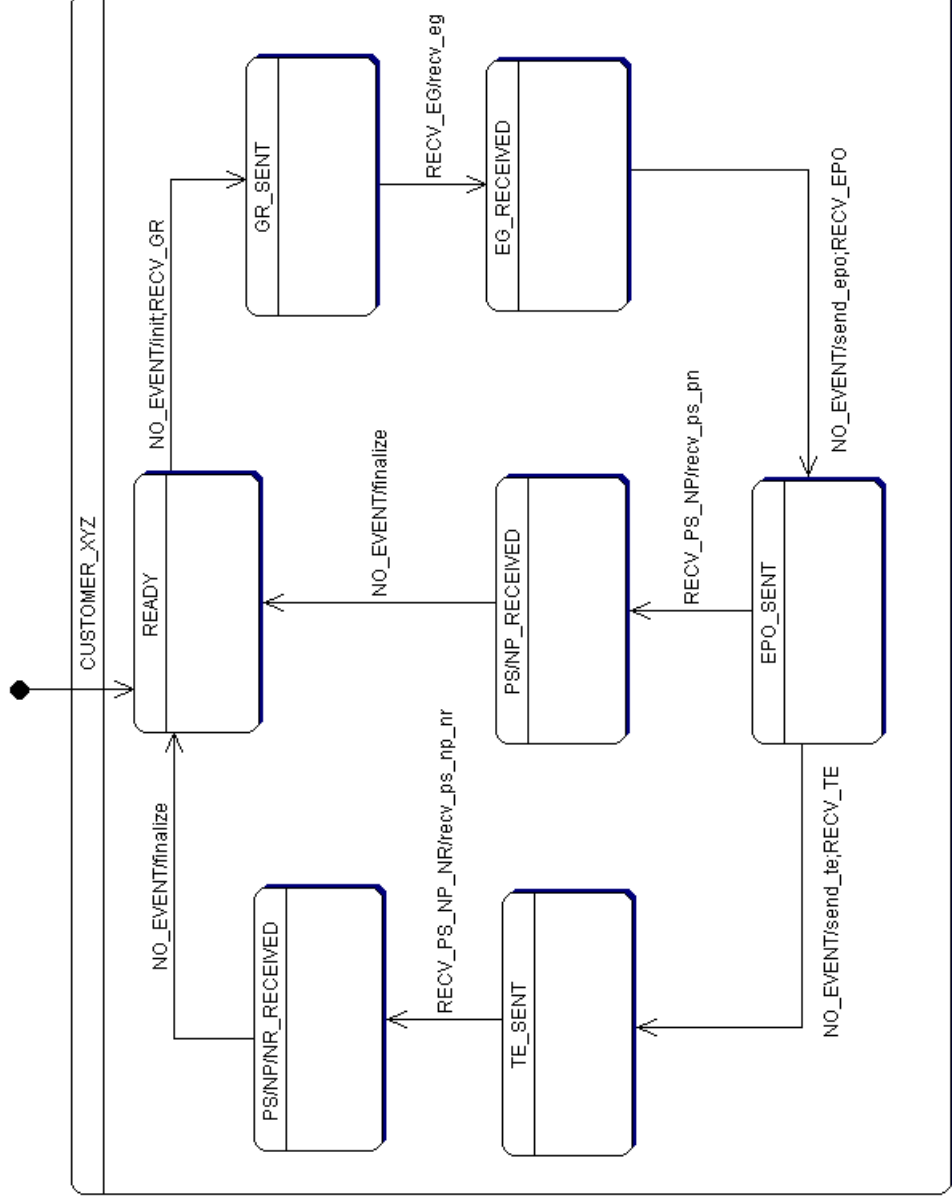
Formalisierung der Idee mit Statecharts



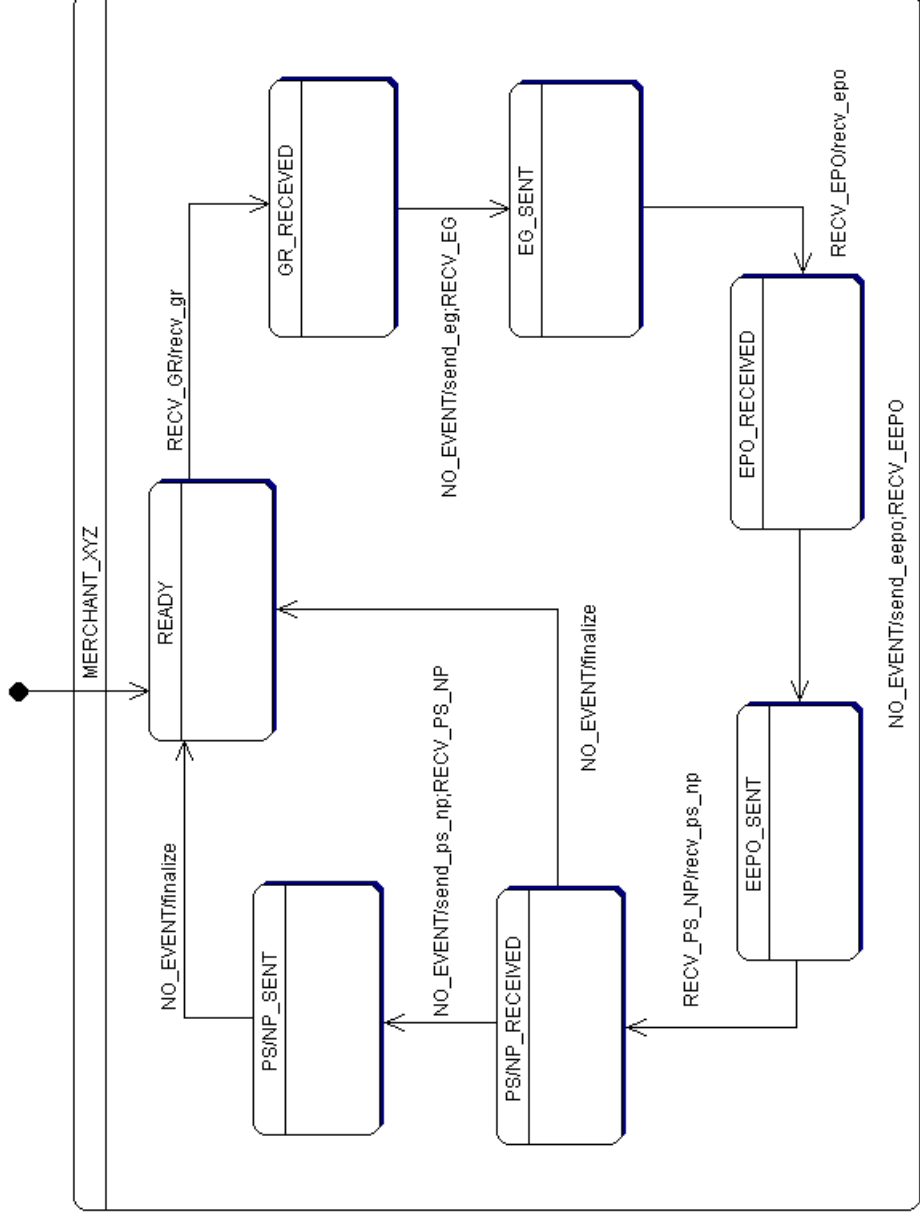
Verfeinerung von Customer-Statechart



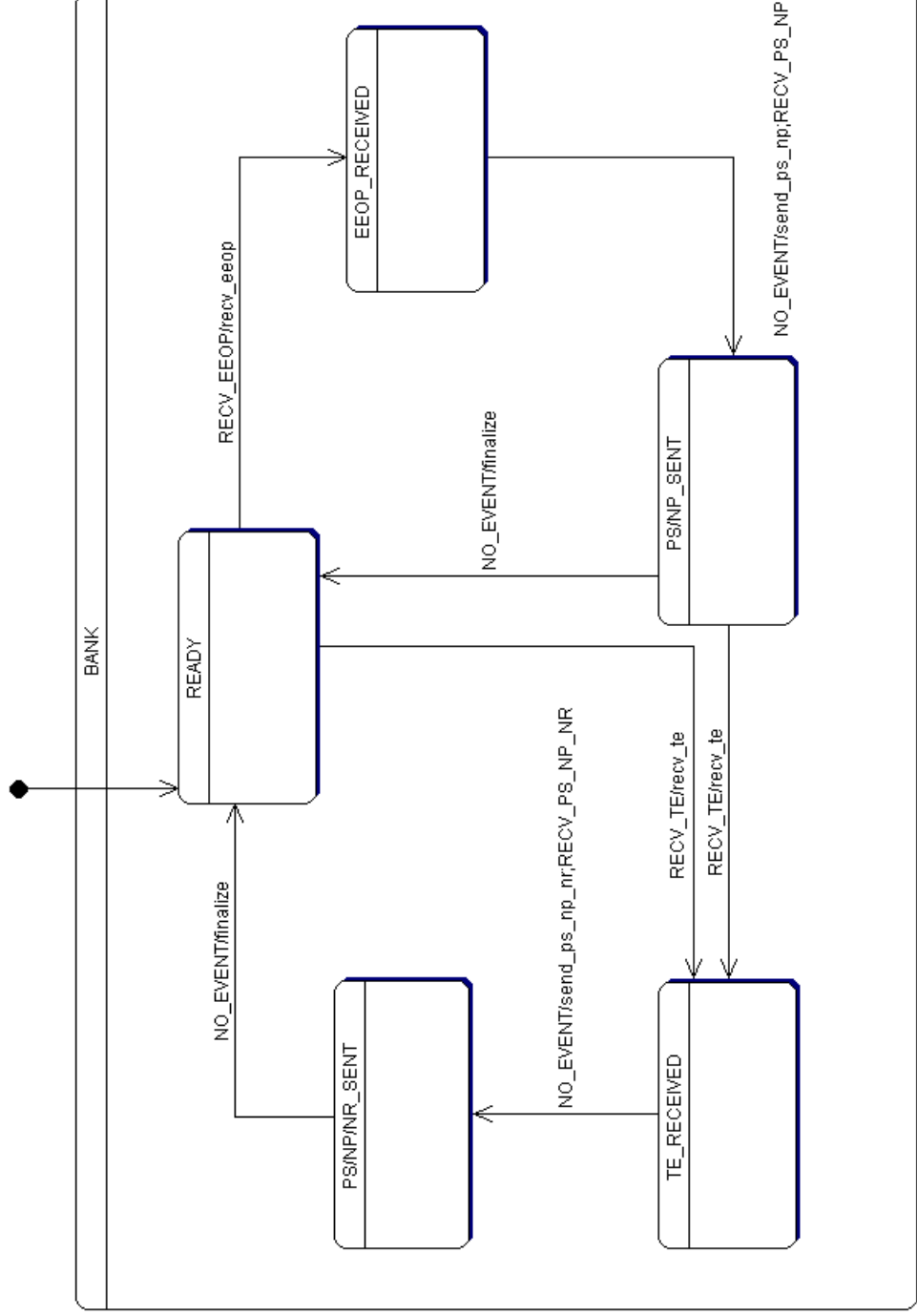
Verfeinerung von Customer-Statechart



Verfeinerung von Merchant-Statechart



Verfeinerung von Bank-Statechart



Zusammenfassung

Vorteile:

- Kompakte graphische Darstellung bei großen Zustandsräumen
- Inkrementelle Vorgehensweise beim Entwurf verteilter Systeme (Top-Down, Bottom-Up)
- Parallele und sequentielle Dekomposition

Nachteile:

- Feste Struktur (keine Änderungen zur Laufzeit)