



Konsistente Verknüpfung von Aktivitäts-, Sequenz- und Zustandsdiagrammen

Darstellungsunabhängige und formale Semantik zur Verhaltensbeschreibung von Echtzeit-Systemen

Dipl.-Inform. Lars Ebrecht



Deutsches Zentrum
für Luft- und Raumfahrt e.V.
in der Helmholtz-Gemeinschaft

Konsistente Verknüpfung von Aktivitäts-, Sequenz- u. Zustandsdiagrammen >
Mobilität und Echtzeit 2007 > 06.12.2007 > 1

Institut für Verkehrsführung und Fahrzeugsteuerung > Technologien aus der Luft- u. Raumfahrt für die Straße und Schiene



Inhalt - Überblick

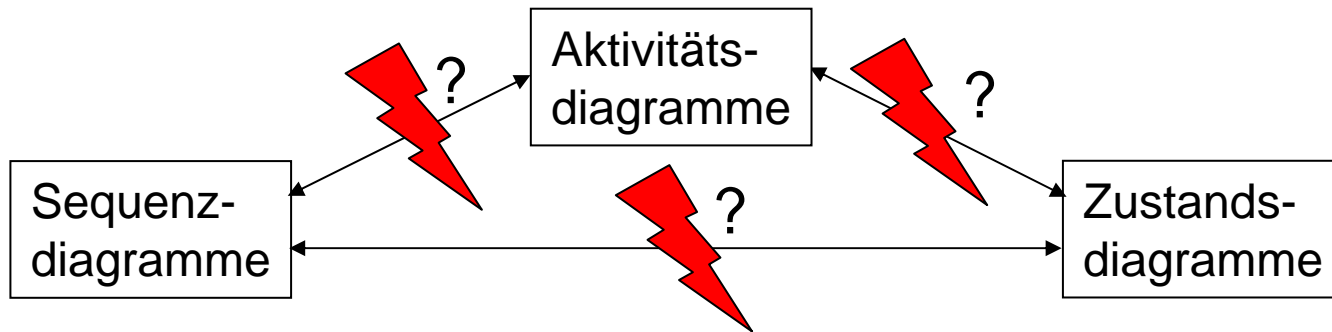
- Motivation / Ziel / Problemstellung
- Darstellungsunabhängige und formale Semantik für das Verhalten von Echtzeit-Systemen
- Konsistente Verknüpfung und Beschreibung von Echtzeit-Systemen mit Aktivitäts-, Sequenz- und Zustandsdiagrammen
- Anwendungsbeispiel
- Zusammenfassung



Motivation / Ziel

- Wie kann das komplizierte und komplexe Verhalten von Echtzeit-Systemen beschrieben werden?
- Anforderungen für Verhaltensbeschreibung von Echtzeit-Systemen
 - möglichst intuitiv, d.h. verständlich und übersichtlich durch grafische Darstellung (mit verschiedenen Ansichten)
 - konsistent, durch generische Struktur für Echtzeit-Systemverhalten und formale und mathematische Basis
- Wie lässt sich ein und dasselbe System aus verschiedenen Perspektiven beschreiben (Top-down, hierarchisch und iterativ)!?

Problemstellung



- UML + Profile
 - Verhaltensdiagrammvielfalt verkompliziert Systembeschreibung
- Vergleich von Aktivitäts-, Sequenz- und Zustandsdiagrammen:

Zustände optional,	ohne Zustände u.	explizite Zustände,
Kontrollflussüberblick/-	keine absolute Zeit,	versteckte Aktionen,
varianten schwierig	Ereignisse optional	Aktivitäten

 - UML Verhaltenssemantik unzureichend
 - keine einheitliche, übergreifende Verhaltenssemantik

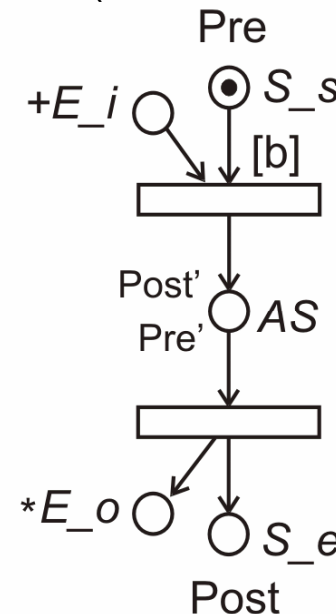
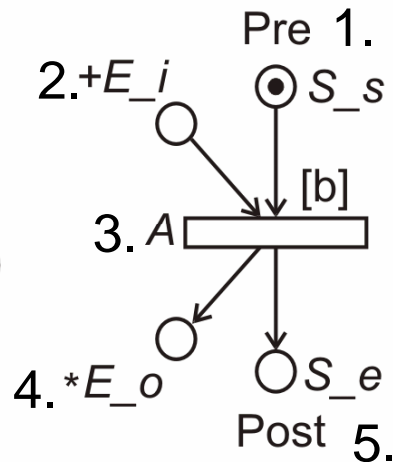
=> Inkonsistente Systemverhaltensbeschreibung möglich!
- Kann ein komplexes Echtzeit-System allein mit einem Diagrammtyp ausreichend und überschaubar beschrieben werden?

Darstellungsunabhängige und formale Semantik für das Verhalten von Echtzeit-Systemen

Wesentliche Elemente des Systemverhaltens:

- Funktionen u. Prozesse (Aktionen/Aktivitäten)
- Start- u. Endzustand (globale und lokale Systemzustände)
- Trigger- u. ausgelöste Ereignisse (extern und intern)
- Generische Verhaltensstruktur/-semantik (Petri-Netz-Darstellung)

Funktion
(kompakt)

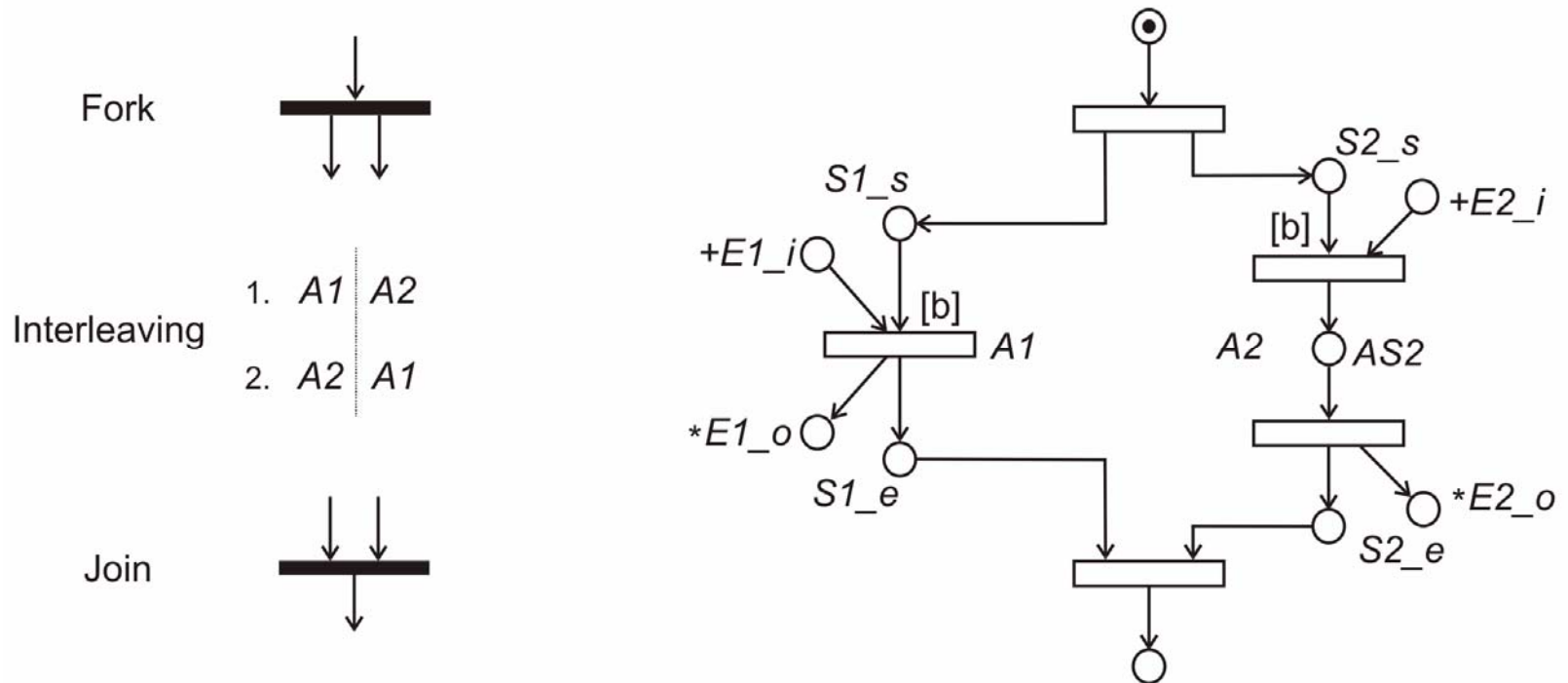


Funktion
(entfaltet)

Darstellungsunabhängige und formale Semantik für das Verhalten von Echtzeit-Systemen

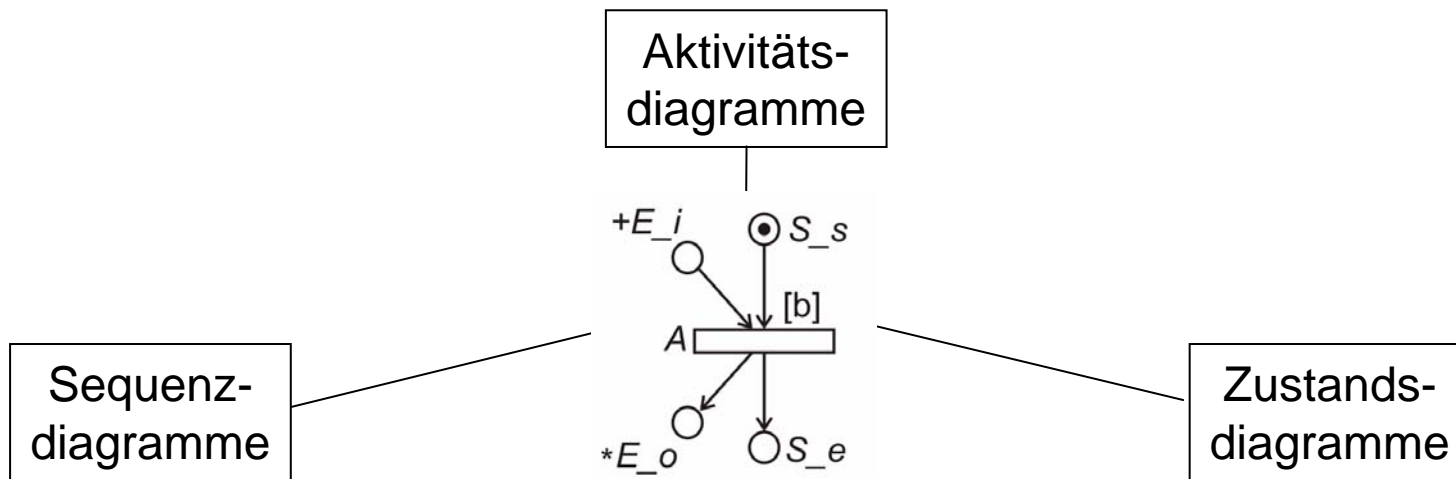
Zusätzlich wichtige Aspekte des Echtzeit-Systemverhaltens:

- Zeit (Rechtzeitigkeit, d.h. nicht vor/nach t oder innerhalb Intervall $t1, t2$)
- Nebenläufigkeit (Fork und Join)



Basis für konsistente Verknüpfung von Aktivitäts-, Sequenz- und Zustandsdiagrammen

- Darstellungsunabhängige Semantik für Echtzeit-Systemverhalten (Aktionen = Aktivitäten, Funktionen oder Prozesse)



=> Durchgängige (d.h. hierarchische und iterative), konsistente Verhaltensbeschreibung von Echtzeit-Systemen

Anwendungsbeispiel

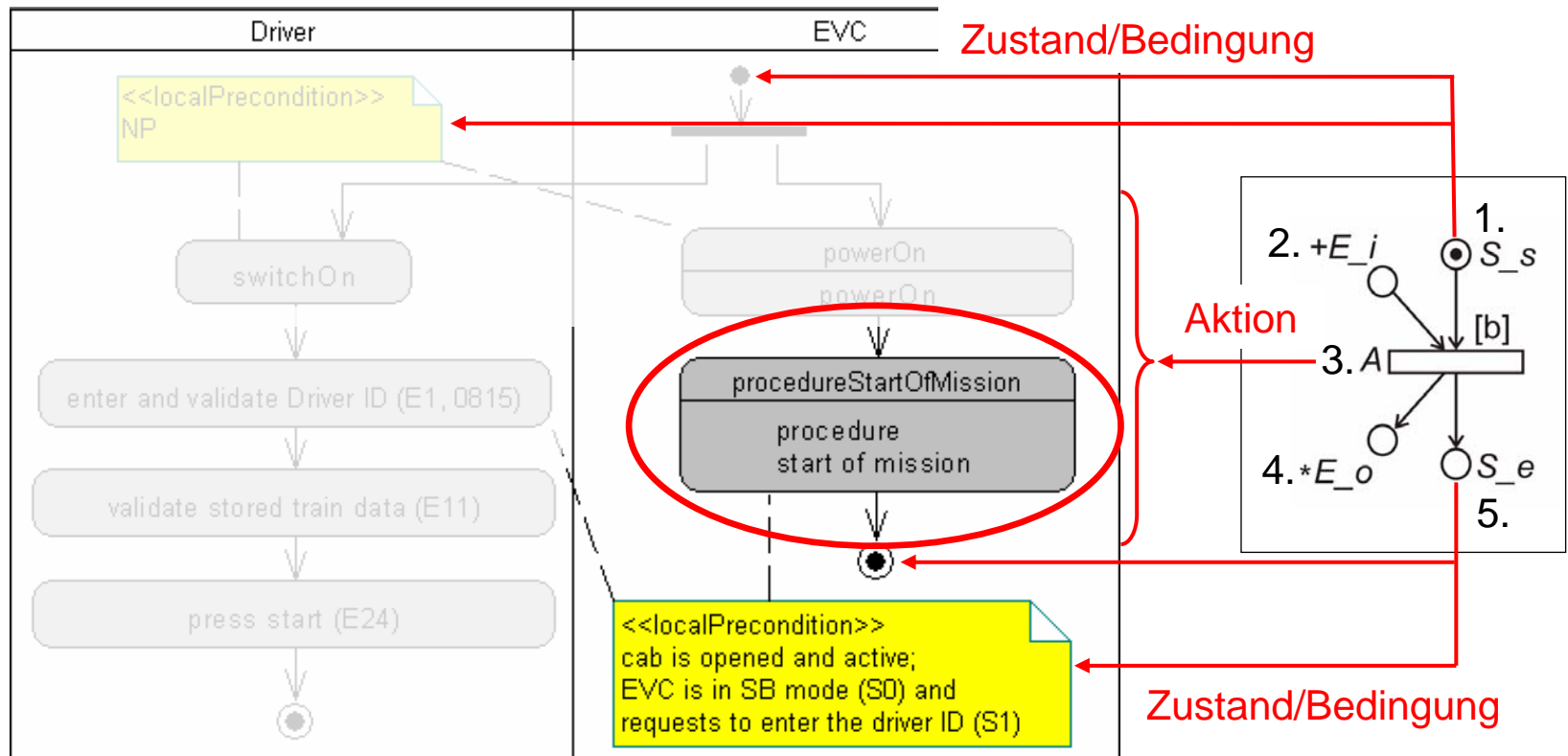
Leit- und Sicherungstechnikkomponente des European Train Control System (ETCS)



ETCS – European Train Control System
EVC – European Vital Computer

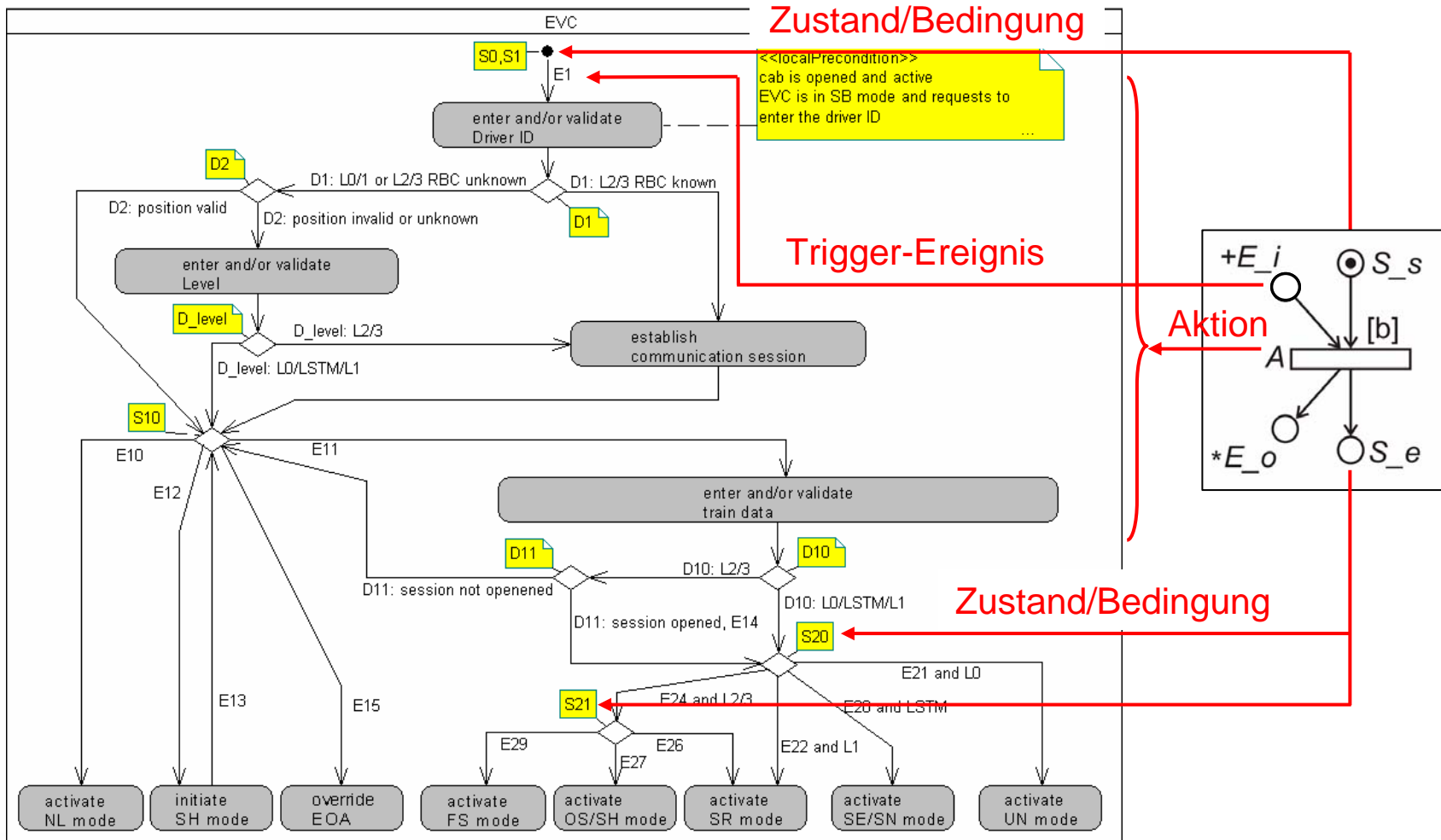


UML Aktivitätsdiagramm – Aktivitäten/Prozesse im Überblick (grob)

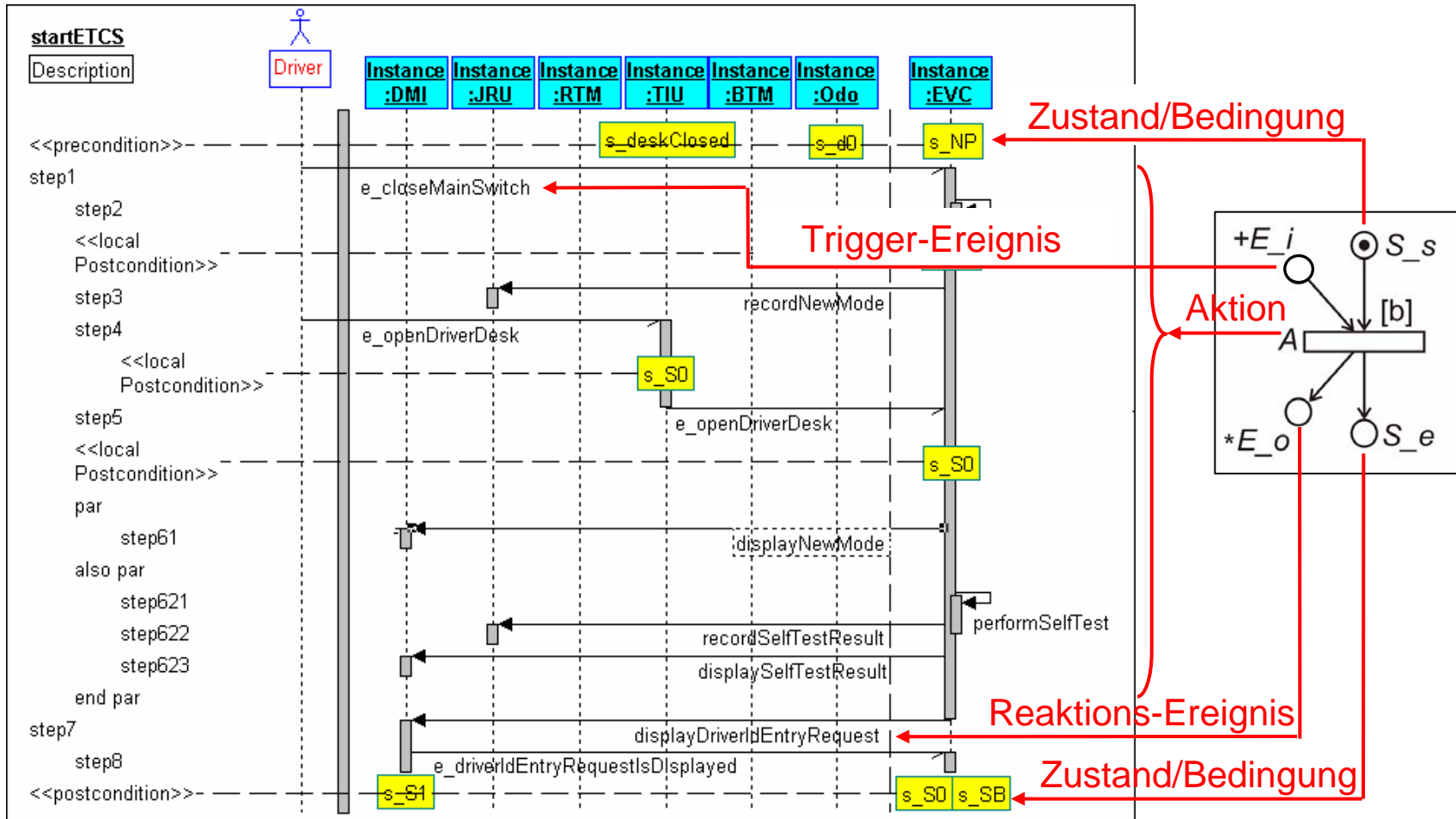


Aktion = Aktivität, Funktion, Prozess

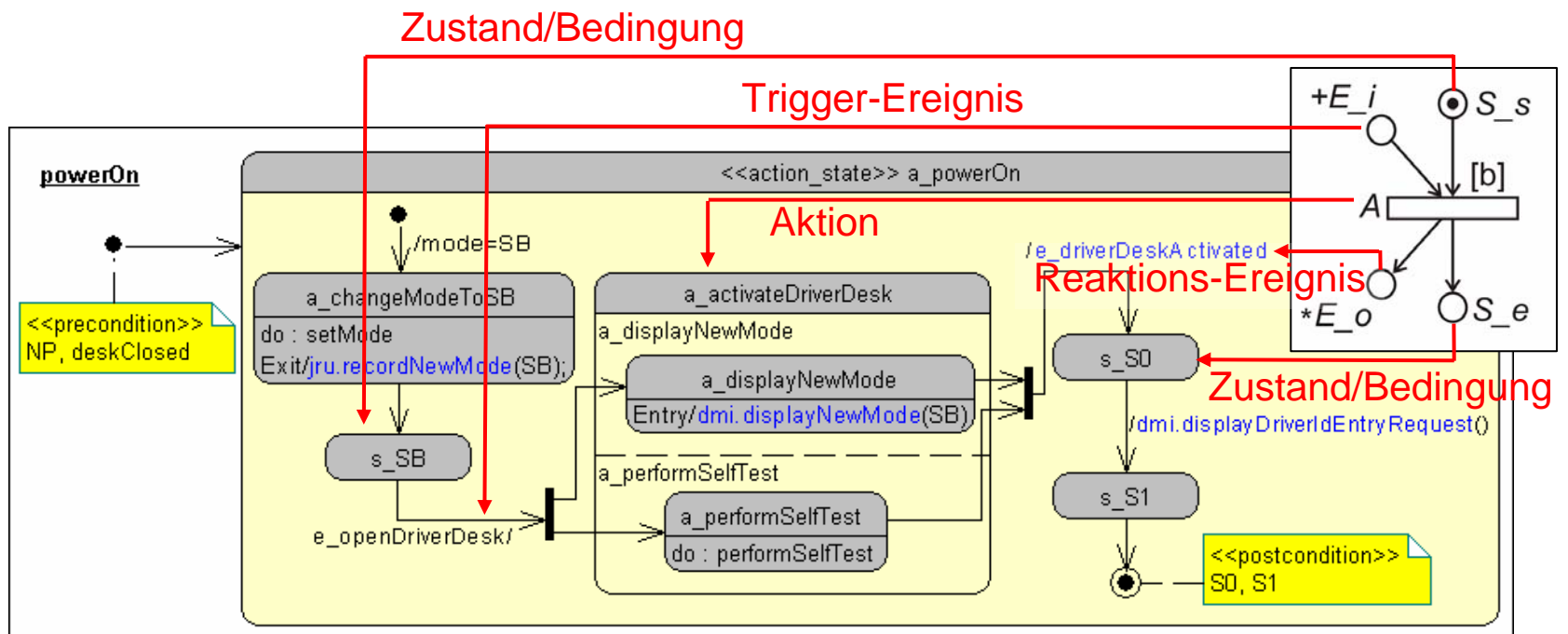
UML Aktivitätsdiagramm – Aktivitäten/Prozesse im Überblick (detailliert)



UML Sequenzdiagramm – Zeitliche Schnittstelleninteraktion im Detail



UML Zustandsdiagramm – Aktivitäten/Prozesse im Detail



Zusammenfassung

- Verwendung von UML Aktivitäts-, Sequenz- und Zustandsdiagramme für differenzierte Beschreibung/Betrachtung ein und desselben Echtzeit-Systemverhaltens
 - Aktivitäten u. Prozesse im Überblick (Kontrollfluss)
 - Zeitliche Schnittstelleninteraktion im Detail (Datenfluss)
 - Aktivitäten/Prozesse im Detail (Zustandstransitionen)
- Darstellungsunabhängige Verhaltensstruktur
 - unterstützt hierarchische und iterative Verhaltensbeschreibung
 - verspricht Identifikation von Inkonsistenzen in Verhaltensbeschreibungen von Echtzeit-Systemen



Vielen Dank für Ihre Aufmerksamkeit!

Fragen, Anmerkungen, Meinungen,
Kritik, Verbesserungen, Anregungen ...!?

Kontakt: Lars.Ebrecht@dlr.de