

# Flexible Softwareentwicklung für Flugsimulatoren in der Forschung am Beispiel des Projektes Lubeta

Jan Hettwer

DLR - Institut für Flugsystemtechnik

Workshop der ASIM/GI-Fachgruppen STS und GMMS

*Simulation technischer Systeme und Grundlagen und Methoden in Modellbildung und Simulation*



Wissen für Morgen

# Besonderheiten von Flugsimulatoren in der Forschung

- Verwendung für unterschiedlichste Projekte
- Simulatornutzung von Entwicklern und Anwendern gleichermaßen
- Vermehrter Einsatz von „In-House“ Software
- Uneinheitliche Softwarestrukturen
- Häufige Hardwareanpassungen
- Einsatz neuer Technologien



# Motivation

- Erhöhung der Flexibilität bei der Softwareentwicklung
  - Experimentalzeit im Simulator verlängern
  - Zufriedenheit der Nutzer eines Simulators steigern
  - Reduzierung des Verwaltungsaufwandes im operationellen Betrieb des Simulators



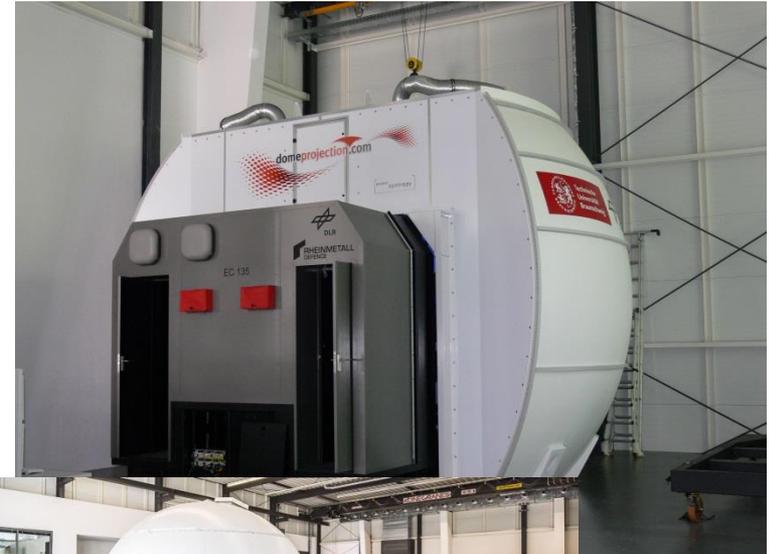
# Flexibilität bei der Softwareentwicklung

- Vereinheitlichung der Softwarestrukturen
- Entwicklung und Testen unabhängig vom Simulator
- Software so gestalten, dass sie modularisiert werden kann
- Konfigurierbarkeit einzelner Module ermöglichen
- Lauffähigkeit verschiedener Konfigurationen sicherstellen
- Emulation von Komponenten vorsehen
- Nutzung einer gemeinsamen Versionsverwaltung



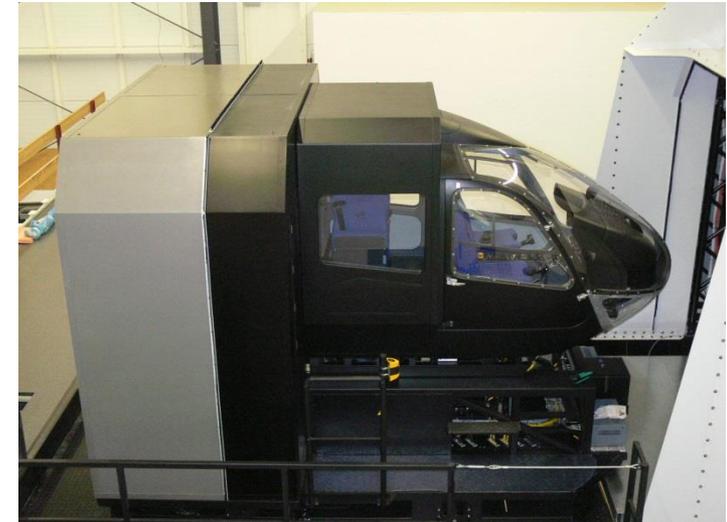
# Der Air Vehicle Simulator AVES

- DLR betreibt am Standort Braunschweig den Air Vehicle Simulator AVES
- Festsitzsimulator
- Bewegungssimulator mit elektropneumatischem 14 t Hexapod-System
- RoRo Wechselsystem zur Nutzung verschiedener Cockpits
- Hochauflösende 15 Kanal Projektionssysteme



# Die AVES Cockpits

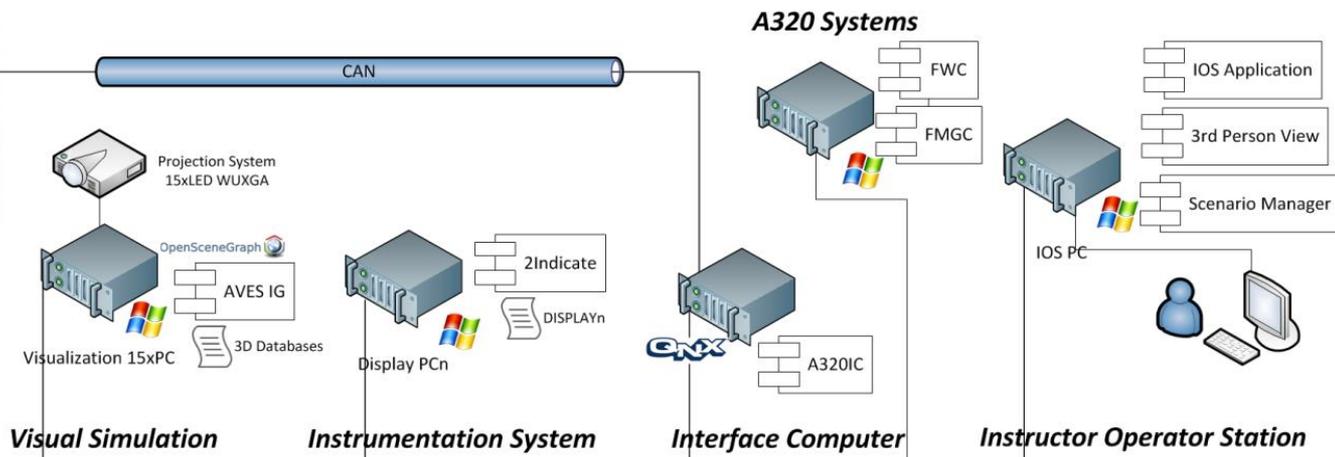
- Airbus A320 Flugzeugcockpit
- Airbus Helicopters EC135 Hubschraubercockpit
- Originalgetreue Replika der Flugversuchsträger
- Cockpitqualität Level D
- Ende 2017 Generische Flugzeugkabine



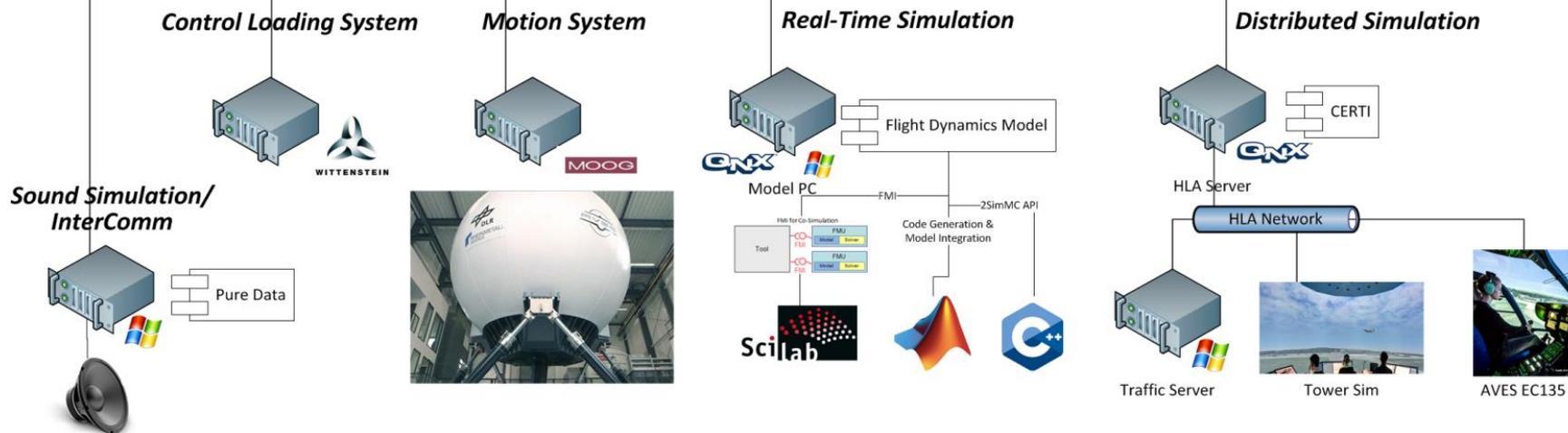
# Die AVES Infrastruktur (A320)



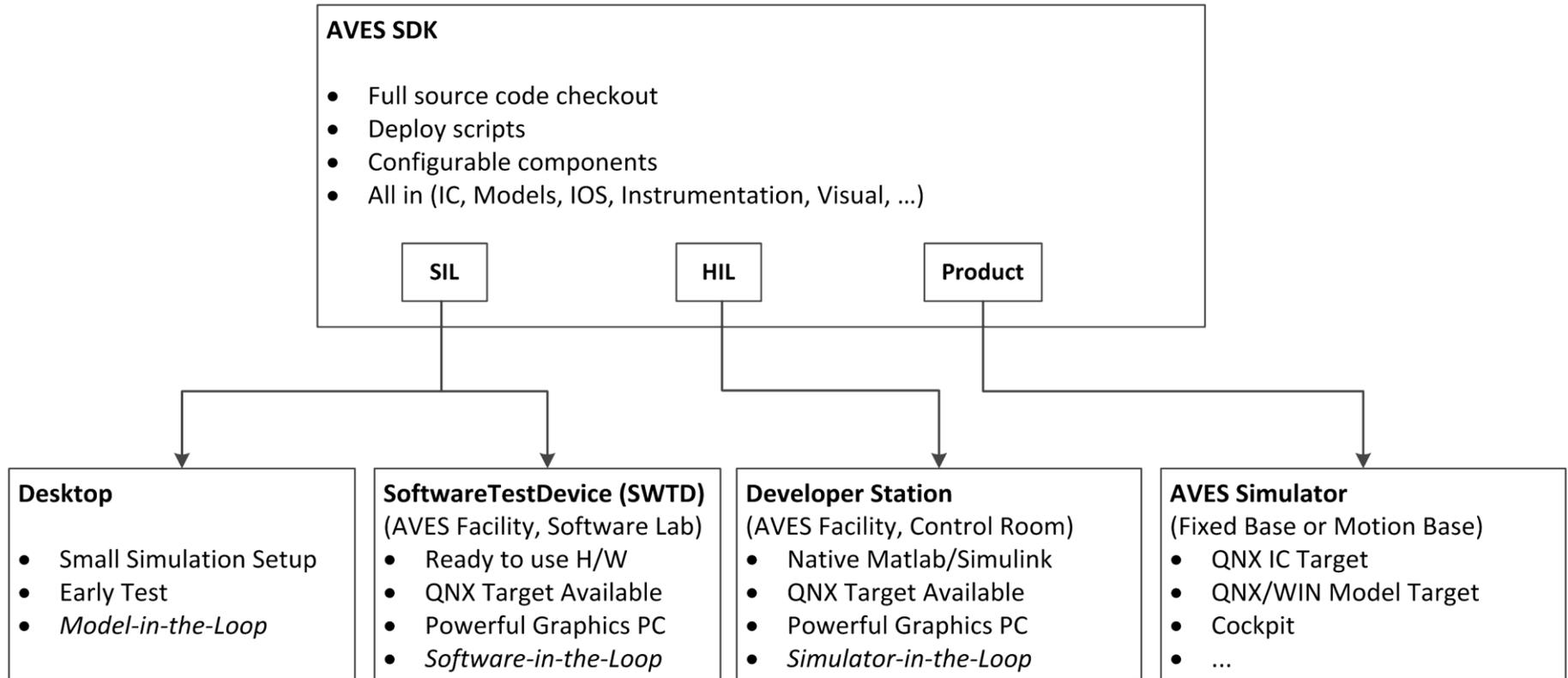
AVES A320 ATRA Cockpit



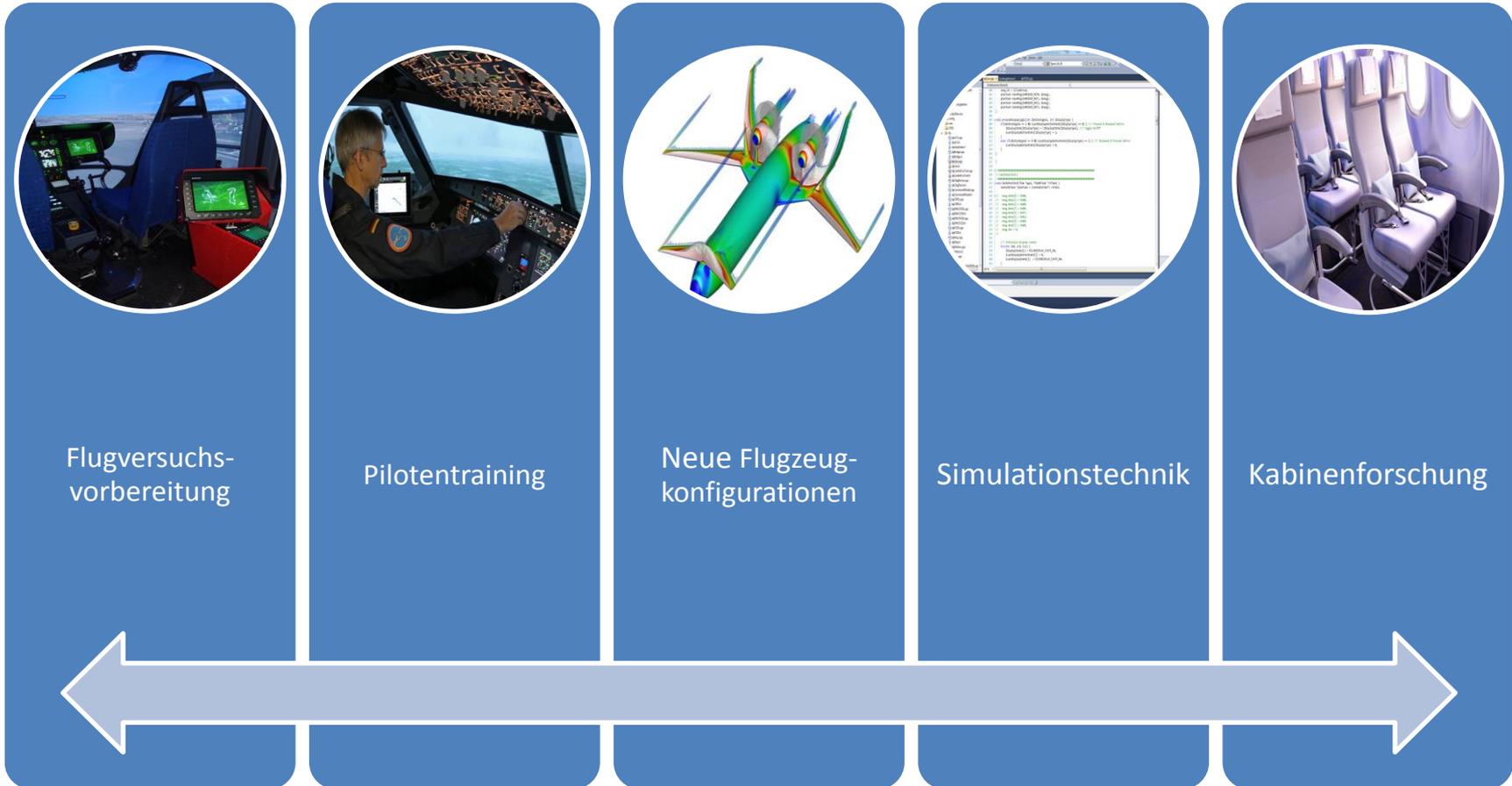
## A320 Systems



# Das AVES Software Development Kit



# Die AVES Forschungsthemen



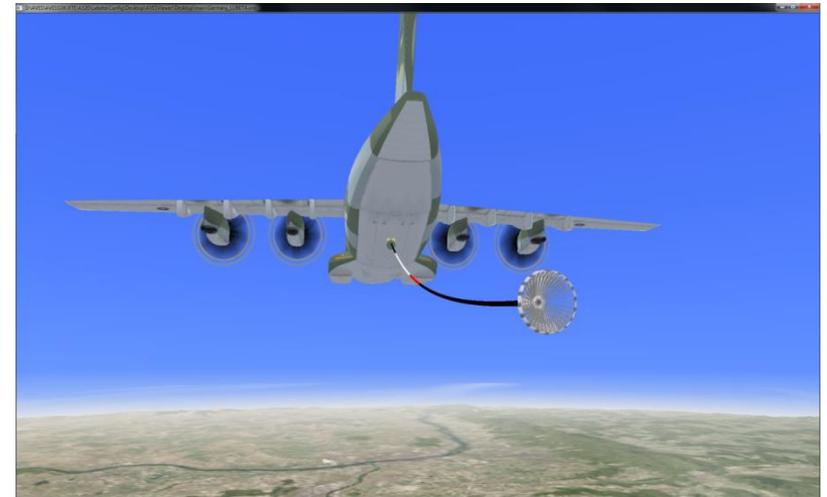
# Das Projekt Lubeta

- Untersuchung der Vorteile des automatisierten Luftbetankungsvorgangs von bemannten und unbemannten Flugzeugen
- Future Military Transport Aircraft
- Simulation eines Tankflugzeuges
- Simulation eines Empfängerflugzeuges
- Modellierung der Betankungshardware
- Modellierung der Betankung
- Implementierung einer automatischen Regelung



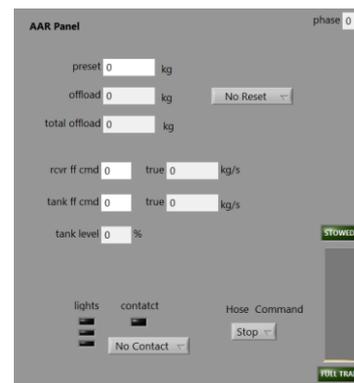
# Der Luftbetankungsvorgang

- Tanker fliegt kontrolliert vom Autopiloten, in definiertem Flugzustand vorweg
- Tankschlauch mit Fangkorb wird ausgefahren
- Receiver nähert sich mit Tanksonde
- Tanksonde wird in Fangkorb eingeführt
- Treibstoff wird vom Tanker zum Receiver gepumpt
- Verbindung wird nach erfolgreicher Betankung gelöst
- Schlauch und Korb werden eingefahren



# Flexibilität im Projekt Lubeta

- Lubeta kann in unterschiedlichen Konfigurationen genutzt werden
- Nutzung im AVES Simulator
  - Operator Station dient der Steuerung des Tankers und der Betankungseinheit
  - A320 Cockpit dient der Steuerung des Receivers
  - Instructor Station dient zusätzlich zur Visualisierung der Außensicht

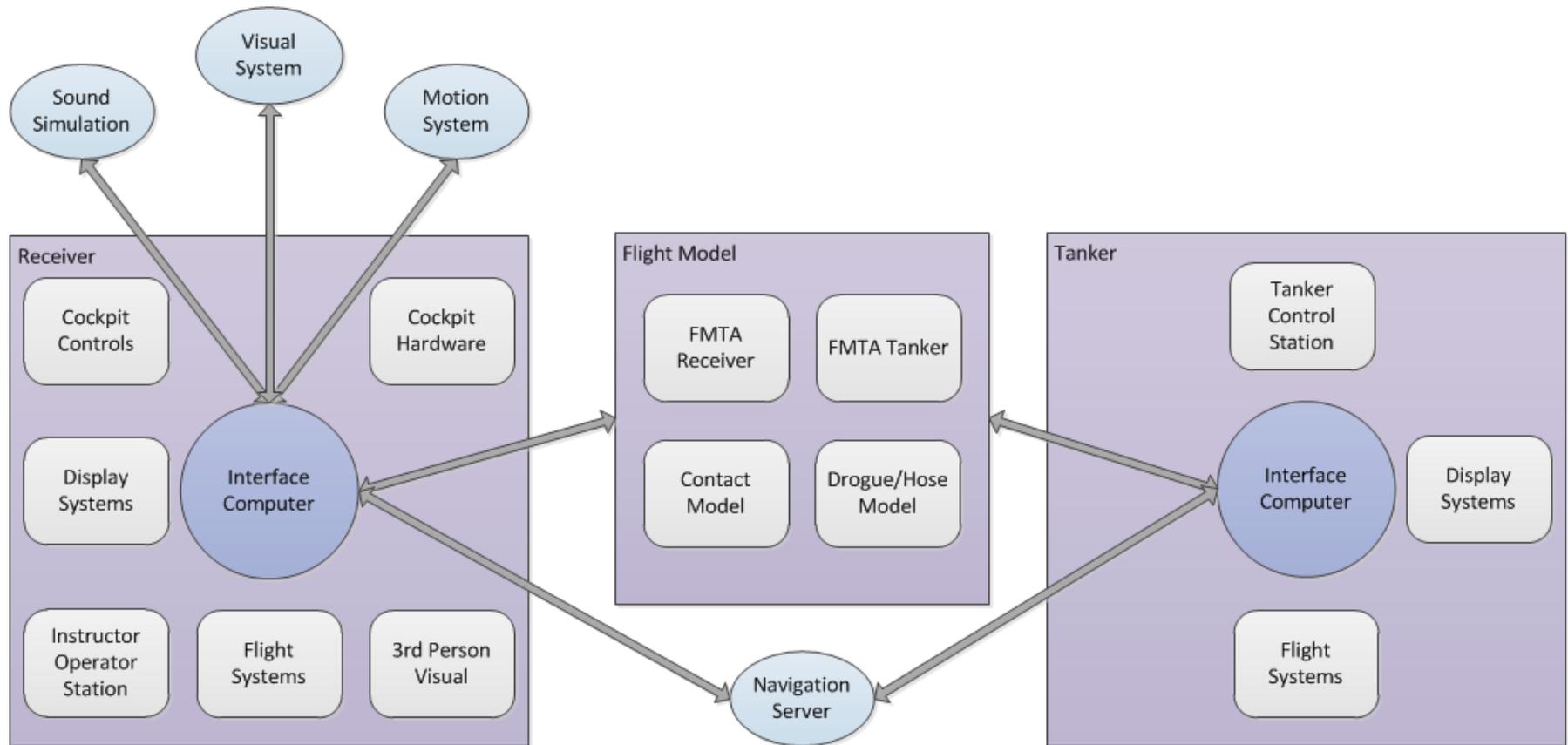


# Flexibilität im Projekt Lubeta

- Nutzung auf dem Software Test Device oder also Desktop Version auf dem Arbeitsplatzrechner



# Die Lubeta Softwareinfrastruktur



# Flexibilität im Projekt Lubeta

- Die Module nutzen einheitlich die Struktur des AVES SDK
- Alle Softwarekomponenten sind modular aufgebaut
- Die Komponenten sind konfigurierbar, ggf. existieren konfigurationsspezifische Versionen
- Essentielle Softwareteile können emuliert werden
  - Controls für den Receiver optional auch für den Tanker, Flight Control Unit
- Nutzung der Lubeta-Simulation ohne personelle Hilfe
- **Verfügbarkeit des Simulators zur Durchführung von Experimenten wird wesentlich erhöht**



# Zusammenfassung und Ausblick

- Das Projekt Lubeta nutzt die verschiedenen Konfigurationen sowohl für die Entwicklung als auch für das Testen von Softwarekomponenten
- Die Lubeta Software kann somit unabhängig vom Simulator im Gesamtkontext weiterentwickelt und betrieben werden
- Verbesserungsmöglichkeiten bestehen beim Handling der Gesamtsoftware in der Desktopversion
- Nächste Schritte:
  - Implementierung eines Reglers für den vollautomatischen Betankungsvorgang
  - Implementierung eines realistischeren Kontaktmodells



**Vielen Dank für Ihre  
Aufmerksamkeit!**



**Gibt es Fragen?**

Jan Hettwer, [jan.hettwer@dlr.de](mailto:jan.hettwer@dlr.de)

Deutsches Zentrum für Luft und Raumfahrt e.V.  
Institut für Flugsystemtechnik  
Lilienthalplatz 7  
38108 Braunschweig

