



Bundesministerium
für Wirtschaft
und Technologie



Gefördert vom Raumfahrtmanagement des Deutschen Zentrums für Luft- und Raumfahrt e.V. mit Mitteln des Bundesministeriums für Wirtschaft und Technologie aufgrund eines Beschlusses des Deutschen Bundestages unter dem Förderkennzeichen **50 RA 1015** und **50 RA 1016**.



Schlußbericht (per 31.12.2012)

nach Anlage 2 zu Nr. 8.2 NKBF 98

"Mission Control Concepts for Robotic Operations"

Förderkennzeichen: 50 RA 1015 und 50 RA 1016

Version: 1.0

Date: 2013-04-22

Reference: VCS-MICCRO-MGT-PR-00049

Zuwendungsgeber



DLR Raumfahrtmanagement

Deutsches Zentrum für Luft- und Raumfahrt (DLR)

Königswinterer Str. 522-524

53227 Bonn-Oberkassel

Ansprechpartner: Herr Thomas Wolf

Email: t.wolf@dlr.de

Verbundpartner



SCISYS Deutschland GmbH

Borgmannstrasse 2

44894 Bochum

Ansprechpartner: Herr Dr. Markus Plura

Tel: +49 234 9258-244

Fax: +49 234 9258-190

Email: markus.plura@scisys.de



Institut für Robotik und Mechatronik

Münchener Straße 20

82234 Weßling

Ansprechpartner: Herr Martin Stelzer

Tel: +49 (8153) 28 2153

Fax: +49 (8153) 28 1134

Email: martin.stelzer@dlr.de



Deutsches Raumfahrt-Kontrollzentrum (GSOC)

Münchener Straße 20

82234 Weßling

Ansprechpartner: Herr Andreas Ohndorf

Tel: +49 (8153) 28 2772

Email: andreas.ohndorf@dlr.de



Unterschriftenliste

	Name	Function	Signature	Date
Author	Dr. Markus Plura	Projektleiter		2013-04-22
Author	Martin Stelzer	Projektleiter		2013-04-22
Approval	Oliver Harrmann			2013-04-22

Tabelle 1 – Unterschriftenliste

Überarbeitungsliste

Issue/Revision	Date	Author	Software Version	Changed Pages/Paragraphs
1.0	22.04.2013	MPL/AOH/MAS		Erstversion

Tabelle 2 – Überarbeitungsliste

Verteiler

Name	No. Copies
SCISYS Archiv	1

Tabelle 3 – SCISYS Interne Verteilerliste

Company/Organisation	Name	No. Copies
DLR-Raumfahrtmanagement	T. Wolf	1
DLR-RMC	M. Stelzer	1
DLR RB-MB (Deutsches Raumfahrtkontrollzentrum)	A. Ohndorf	1

Tabelle 4 – SCISYS Externe Verteilerliste

Inhaltsverzeichnis

1	Einleitung.....	9
1.1	Zielgruppe.....	9
1.2	Konventionen.....	9
1.3	Mitgeltende Dokumente.....	9
1.4	Referenzdokumente	10
2	Projektübersicht.....	11
2.1	Aufgabenstellung.....	11
2.2	Voraussetzungen für die Vorhabensdurchführung.....	11
2.3	Planung und Ablauf des Vorhabens	11
2.4	Wissenschaftliche und technische Vorarbeiten.....	13
2.4.1	Vorarbeiten der SCISYS Deutschland GmbH.....	13
2.4.2	Vorarbeiten des DLR-RMC.....	16
2.4.3	Vorarbeiten des DLR-GSOC	17
2.5	Zusammenarbeit mit anderen Stellen	18
3	Projektergebnisse.....	19
3.1	Erzielte Ergebnisse.....	19
3.1.1	Missionsbetriebskonzept für robotische Missionen	20
3.1.2	Systemdemonstrator	27
3.1.3	Systemverifikation	45
3.2	Nutzen und Verwertbarkeit der Ergebnisse.....	55
3.3	Fortschritte auf dem Gebiet des Vorhabens.....	55
3.4	Veröffentlichungen der Projektergebnisse	56
3.5	Wichtigsten Positionen des zahlenmäßigen Nachweises	56
3.6	Notwendigkeit und Angemessenheit der geleisteten Arbeit.....	56
4	Erfolgskontrollbericht.....	57
5	Kurzfassung	57

Abbildungsverzeichnis

Abbildung 1: Projektmanagementplan (Meilensteine).....	12
Abbildung 2: Projektmanagementplan (Projektmanagement).....	12
Abbildung 3: Projektmanagementplan (Technische Arbeitspakete)	12
Abbildung 4: Strukturierung des Vorhabens.....	13
Abbildung 5: Systemarchitektur des SCOS@MAL Adapter.....	14
Abbildung 6: Systemarchitektur des MMOPS (Mars Mission On-Board Planner and Scheduler).....	15
Abbildung 7: CREST Experimentaufbau	16
Abbildung 8: GETEX-Experiment an Bord des ETS-7-Satelliten	16
Abbildung 9: ROKVISS-Manipulator	17
Abbildung 10: Die EPOS-Anlage am DLR-GSOC (Stand 2009).....	18
Abbildung 11: Organisationsstruktur im Kontrollraum	21
Abbildung 12: Abbildung des Raumsegmentes auf das Bodensegment	22
Abbildung 13: Verlängerung der Kontaktzeit durch ein Netzwerk aus Bodenstationen.....	23
Abbildung 14: Implementierung virtueller Kommunikationskanäle.....	24
Abbildung 15: Kommunikationsstruktur	24
Abbildung 16: Benutzerschnittstelle	26
Abbildung 17: End-to-End-Systemdemonstrator	27
Abbildung 18: Hardwarekomponenten des Systemdemonstrators	29
Abbildung 19: Aufbau der Netzwerkinfrastruktur und der installierten Plattformen des MICCRO Demonstrators	30
Abbildung 20: Räumliche Aufteilung der Systemkomponenten	31
Abbildung 21: Konsolenzuordnung und Raumaufteilung im Kontrollraum.....	31
Abbildung 22: Komponenten des Robotic Control Systems (RCS)	32
Abbildung 23: Komponenten der Benutzerschnittstelle	33
Abbildung 24: Ausschnitte aus der Problem- und Umgebungsbeschreibung für eine telepräsen- te OOS-Operation in PDDL	34
Abbildung 25: Orchestrierung des Planungsprozesses durch den RCS/Planning Server.....	35
Abbildung 26: Mechanismen zur Steuerung des Planungsprozesses.....	36
Abbildung 27: Empfang von SCOS basierter Telemetrie im egmc ² -Framework.....	37
Abbildung 28: Simulationsplattform mit allen relevanten Schnittstellen während der Integration bei SCISYS in Bochum	37
Abbildung 29: Getrennte Plattformen für die Ansteuerung des ROKVISS-Roboters (links) und des Joysticks (rechts) während der Integration bei RMC und GSOC in Oberpfaffenhofen.....	38
Abbildung 30: Komponenten des Mission Control Systems (MCS).....	39
Abbildung 31: Fiktiver MICCRO OOS Servicer	41
Abbildung 32: Schnittstellen des Communication Gateways (CommGW) und des Onboard Multiplexers (Onboard MUX).....	42
Abbildung 33: Protokollstruktur (Onboard MUX → CommGW)	42
Abbildung 34: Protokollstruktur (CommGW → Onboard MUX)	43
Abbildung 35: Design des Robotik Payload Simulators	44
Abbildung 36: Schematischer Ablauf eines Satellitenkontakts	47
Abbildung 37: Pass 1: Bus Only Operation	48
Abbildung 38: Pass 2: Handover to RCS and Initialisation	49
Abbildung 39: Pass 3: Telepresence Operation.....	50
Abbildung 40: Pass 4: Supervised Autonomy and Abort Operation	51
Abbildung 41: Pass 5: Finalisation of Robotic Operations	52



Abbildung 42: Simuliertes Raumsegment im Robotik-Labor des DLR-RMC 54
Abbildung 43: Durchführung des simulierten Missionsbetriebs im Kontrollraum des DLR-GSOC 54
Abbildung 44: Verwendung des MICCRO-Konzeptes für andere Missionen mit robotischen
komponenten 55



Tabellenverzeichnis

Tabelle 1 – Unterschriftenliste	2
Tabelle 2 – Überarbeitungsliste.....	2
Tabelle 3 – SCISYS Interne Verteilerliste	2
Tabelle 4 – SCISYS Externe Verteilerliste	2
Tabelle 5 – Abkürzungsverzeichnis.....	8
Tabelle 6 – Mitgeltende Dokumente.....	9
Tabelle 7 – Referenzdokumente	10

Abkürzungsverzeichnis

Abkürzung	Beschreibung
AOCS	Attitude and Orbit Control Subsystem
AOS	Acquisition of Signal
BMBF	Bundesministerium für Bildung und Forschung
CAM	Collision Avoidance Maneuver
CCSDS	Consultative Committee for Space Data Systems
CMD	Command Operator
COTS	Common off the shelf
CTX	Cortex
DDD	Detailed Design Description
DEOS	Deutsche Orbitale Servicing Mission
DLR	Deutsches Zentrum für Luft- und Raumfahrt
DoD	Depth of Discharge
ECSS	European Cooperation for Space Standardization
EPOS	European Proximity Operations Simulator
ESA	European Space Agency
ESOC	European Space Operations Center
ETS	Engineering Test Satellite
EXIF	External Interface
FD	Flight Director
FDIR	Failure Detection, Isolation, and Recovery
FOT	Flight Operations Team
FRM	Functional Reference Model
GCC	Generic Control Client
GDC	Generic Data Client
GEO	Geostationary Orbit
GETEX	German Technology Experiment
GNC	Guidance, Navigation, and Control
GSOC	German Space Operation Center
HF	High Frequency
HMI	Human Machine Interface
IPS	Intelligent Planning and Scheduling
ISS	International Space Station
JAXA	Japan Aerospace Exploration Agency
LEO	Low Earth Orbit
LOS	Loss of Signal
MAL	Message Abstraction Layer
MARCO	Modulkar A&R Controller
MCS	Mission Control System
MIB	Mission Information Base
MMOPS	Mars Mission On-Board Planner and Scheduler
MOD	Mission Operations Director
MOS	Mission Operations System (Director)
MOT	Mission Operations Team
NASA	National Aeronautics and Space Administration
NASDA	National Space Development Agency
NAT	Network Address Translation
NC	Network Controller
NCTRS	Network Control and Telemetry Routing System
NRT	Non-realtime
OLEV	On-Orbit Life Extension Vehicle
OOS	On-Orbit Servicing
PDDL	Problem Domain Definition Language
PEX	Procedure Execution
PTS	Power and Thermal Subsystem
PUS	Packet Utilisation Standard
RCS	Robotic Control Subsystem
RMC	Robotics and Mechatronics
ROKVISS	Robotik-Komponenten-Verifikation auf der ISS
ROTEX	Robotic Technology Experiment
RT	Realtime
RvD	Rendezvous and Docking
SCOS	Satellite Control and Operation System
SLE	Space Link Extension
SSD	Software Specification Document
STD	Software Transfer Document

Abkürzung	Beschreibung
STL	Satellite (Support) Team Lead
SUM	Software User Manual
SVVP	Software Verification and Validation Plan
TCP/IP	Transmission Control Protocol / Internet Protocol
TM/TC	Telemetry/Telecommand
TN	Technical Note
TRDSS	Tracking and Data Relay Satellite System
TVCR	Timeline Validation, Control, and Repair
UPS	Unified Propulsion Subsystem
WHM	Weilheim (ground station)

Tabelle 5 – Abkürzungsverzeichnis

1 Einleitung

Dieses Dokument stellt den Schlussbericht für das Fördervorhaben "MICCRO: Mission Control Concepts for Robotic Operations" dar.

Der Berichtszeitraum dieses Berichts fällt zeitlich mit einem Abrechnungsquartal und dem Projektende (per 31.12.2012) zusammen.

Das Vorhaben wird gefördert vom Zuwendungsgeber Raumfahrtmanagement des Deutschen Zentrums für Luft- und Raumfahrt e.V. mit Mitteln des Bundesministeriums für Wirtschaft und Technologie aufgrund eines Beschlusses des Deutschen Bundestages unter den Förderkennzeichen **50 RA 1015** und **50 RA 1016**.

Die in diesem Dokument enthaltenen Informationen orientieren sich an der Struktur, wie sie durch das Muster eines Schlussberichtes (Anlage 2 zu Nr. 8.2 NKBF 98) vorgegeben ist.

Als Teil des Berichtswesens und der Managementdokumentation wird dieses Dokument in deutscher Sprache verfasst. Lediglich die aus anderen Dokumenten entnommenen technischen Informationen, sowie die referenzierten technischen Dokumente selbst, sind in Englisch erstellt.

1.1 Zielgruppe

Dieser Bericht wurde verfasst, um diesen dem Zuwendungsgeber DLR Raumfahrtmanagement gemäß den Förderrichtlinien zur Verfügung zu stellen. Weiter werden auch alle Projektpartner den Bericht als Mitautoren verfolgen.

1.2 Konventionen

Zum 12.12.2011 erfolgte die formwechselnde Umwandlung der VCS Aktiengesellschaft (in vorangegangenen Dokumenten auch VCS AG oder VCS genannt) in die SCISYS Deutschland GmbH (auch als SCISYS benannt). Dies wird in diesem Dokument entsprechend berücksichtigt und es wird das Unternehmen einheitlich mit SCISYS benannt. Dies gilt der Einfachheit halber auch für Ereignisse die vor der Umwandlung stattfanden.

1.3 Mitgeltende Dokumente

ID	Dokument	Referenz	Version/Datum
AD-1	MICCRO Vorhabensbeschreibung	4.0	15.06.2010
AD-2	DLR Zuwendungsbescheinigung	50 RA 1015	25.10.2010
AD-3	BMBF Nebenbestimmungen zur Zuwendungsbescheinigung	NKBF 98	04/2006
AD-4	BMBF Hinweise für Zahlungsempfänger	3220/06.06	06/2006

Tabelle 6 – Mitgeltende Dokumente

1.4 Referenzdokumente

ID	Dokument	Referenz	Version/Datum
RD-3	Software Specification Document (SSD)	VCS-MICCRO-TECH-REQ-000012	1.2
RD-4	Detailed Design Document (DDD)	VCS-MICCRO-TECH-DD-000011	1.0
RD-5	Software User Manual (SUM)	VCS-MICCRO-OPS-MAN-000010	1.0
RD-6	Software Verification and Validation Plan (SVVP)	VCS-MICCRO-AIV-PL-000007	1.0
RD-7	Software Transfer Document (STD)	VCS-MICCRO-AIV-RP-000008	1.0
TN4	Mission Operations Concept	VCS-MICCRO-TECH-TN-000004	3.0
TN5	Autonomy Concept	VCS-MICCRO-TECH-TN-000005	3.0
TN6	User Concept	VCS-MICCRO-TECH-TN-000006	3.0
TN18	Communication Concept	VCS-MICCRO-TECH-TN-000018	3.0
TaD	Terms and Definitions	VCS-MICCRO-TECH-TN-000015	1.0
EKB	Erfolgskontrollbericht	VCS-MICCRO-MGT-PR-000017	1.0

Tabelle 7 – Referenzdokumente

2 Projektübersicht

2.1 Aufgabenstellung

Das übergeordnete Vorhabensziel des Forschungs- und Entwicklungsvorhabens ist die Anforderungsanalyse, Erarbeitung und Verifikation eines möglichst universellen Missions- und Betriebskonzeptes zur Unterstützung vielfältiger robotischer Missionsszenarien mit unterschiedlichem Autonomiegrad.

Die Projektergebnisse sollten dem Unternehmen SCISYS Deutschland GmbH helfen eine deutliche Stärkung der technologischen Kompetenz und eine hohe Nutzerakzeptanz zu erzielen. Mit Hilfe von - dem Vorhaben zeitnah nachgeschalteter - Produktentwicklungs- und Vertriebsmaßnahmen sollte eine entsprechende Marktpräsenz der Lösungen erreicht werden. Im Kern wird dies dazu beitragen, im Unternehmen SCISYS Deutschland GmbH Arbeitsplätze zu sichern und zu schaffen.

2.2 Voraussetzungen für die Vorhabensdurchführung

Raumfahrttechnologie ist aus der modernen Industrie- und Informationsgesellschaft des 21. Jahrhunderts nicht mehr wegzudenken. Automation und Robotik kommt dabei eine Schlüsselfunktion zu. Denn die Raumfahrtrobotik ist unverzichtbar für zahlreiche Raumfahrtanwendungen und sie ist darüber hinaus eine Sprungbretttechnologie für Anwendungen auf der Erde; beispielsweise für Anwendungen in der Medizin, der Energieversorgung, der Landwirtschaft, im Katastrophen- und Umweltschutz, der Meerestechnik, der Luftfahrt und dem Verkehr.

Das Bundesministerium für Wirtschaft und Technologie erkennt daher in der Raumfahrtrobotik eine der wesentlichen Herausforderungen für die Raumfahrt des 21. Jahrhunderts. Dabei gilt es, die führende Rolle Deutschlands bei der Fernsteuerung von Weltraumrobotern in der Erdumlaufbahn (Erdorbit) und den Vorsprung im Vergleich zu anderen Mitbewerbern wie den USA, Japan oder auch Korea zu sichern und auszubauen.

Zukünftige unbemannte Weltraummissionen werden verstärkt teleoperativ geprägt sein und erwartungsgemäß auch im kommerziellen Bereich Einzug halten. Die Bandbreite teleoperativer Missionen ist dabei weit aufgespannt. Neben Szenarien wie Landern, Surface Mobility und In-Situ Experimenten sind neuartige Anwendungsbereiche wie On-Orbit Servicing (OOS) zu nennen. Durch die robotische Prägung und aufgrund der Tatsache, dass Phasen unterschiedlichen Autonomiegrades während einer Mission existieren, erwachsen besondere Anforderungen an den Betrieb und die Bodenkontrollenrichtungen. Daneben verlangt der routinemäßige Einsatz standardisierte Ansätze, die einen sicheren, geregelten Betrieb garantieren. Die Ableitung geeigneter Anforderungen und der Entwurf eines universellen Missionskonzeptes trieben das hier beschriebene Vorhaben in Übereinstimmung mit den förderfähigen Schwerpunkten.

Das Vorhaben unterstützt damit die förderpolitischen Ziele der Bundesregierung und die Interessen der mandatierten Fachorganisationen Technologieentwicklungen im Bereich Weltraumrobotik zu fördern und dadurch die Wettbewerbsfähigkeit des Standorts Deutschland weiter zu erhöhen.

2.3 Planung und Ablauf des Vorhabens

Die unterschiedlichen technischen Aufgaben sowie das Projektmanagement wurden für unterschiedliche Arbeitspakete gegliedert und anhand des Projektmanagementplans durchgeführt.

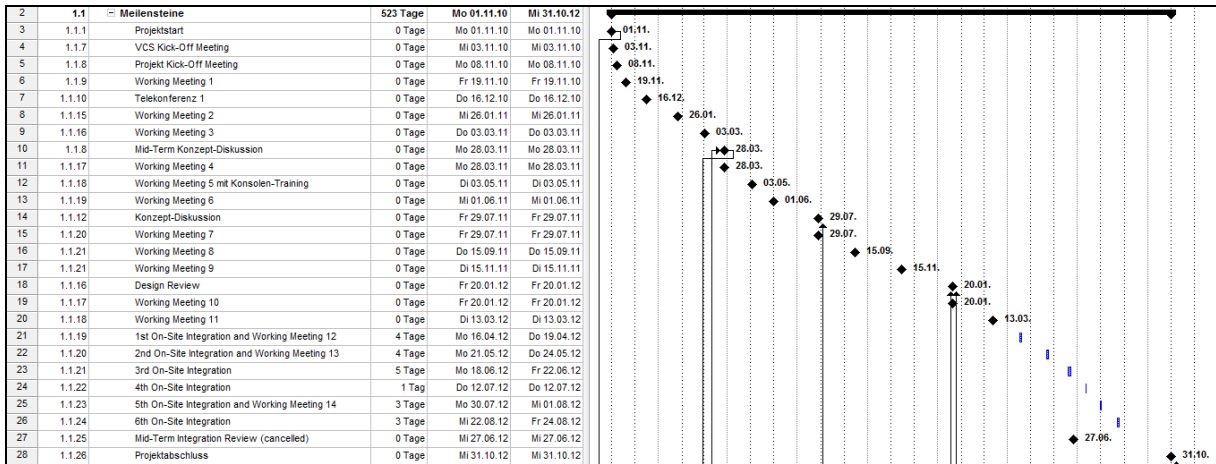


Abbildung 1: Projektmanagementplan (Meilensteine)

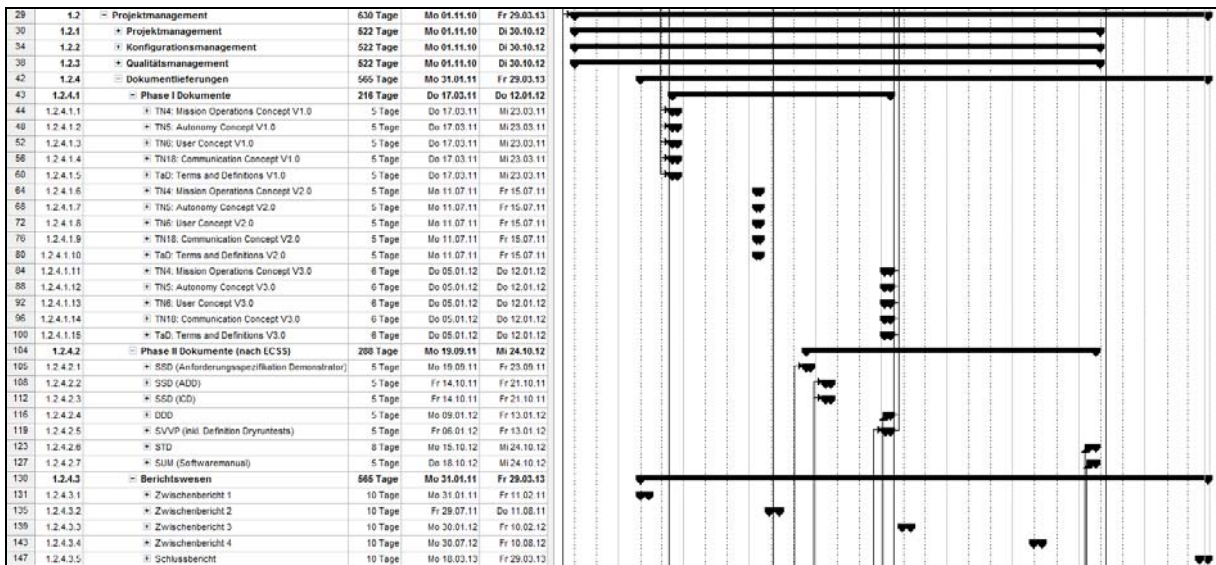


Abbildung 2: Projektmanagementplan (Projektmanagement)

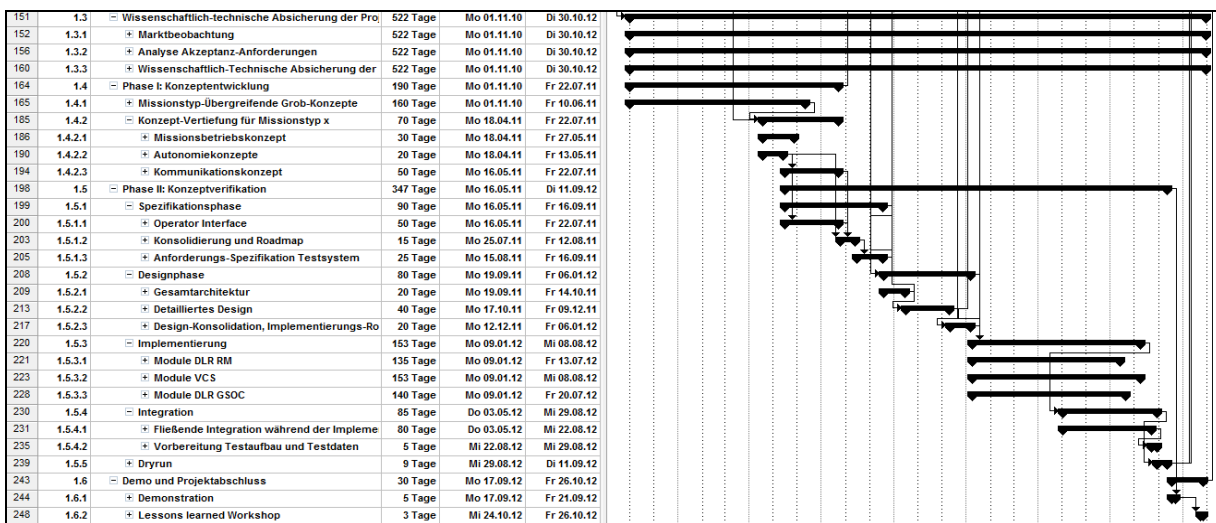


Abbildung 3: Projektmanagementplan (Technische Arbeitspakete)

Für eine detaillierte Beschreibung der durchgeführten Arbeiten in den unterschiedlichen technischen Arbeitspaketen wird hier auf die Zwischenberichte des Vorhabens verwiesen.

Wie im nachfolgenden Diagramm gezeigt, wurde das Vorhaben in zwei Projektphasen untergliedert. Innerhalb der ersten Phase wurden die Konzepte für die Implementierung von robotischen (bzw. OOS-) Raumfahrtmissionen erarbeitet.

In der nachfolgenden Projektphase wurde dieses Konzept exemplarisch im Rahmen eines End-to-End-Systemdemonstrators verifiziert und während einer Demonstration vorgestellt.

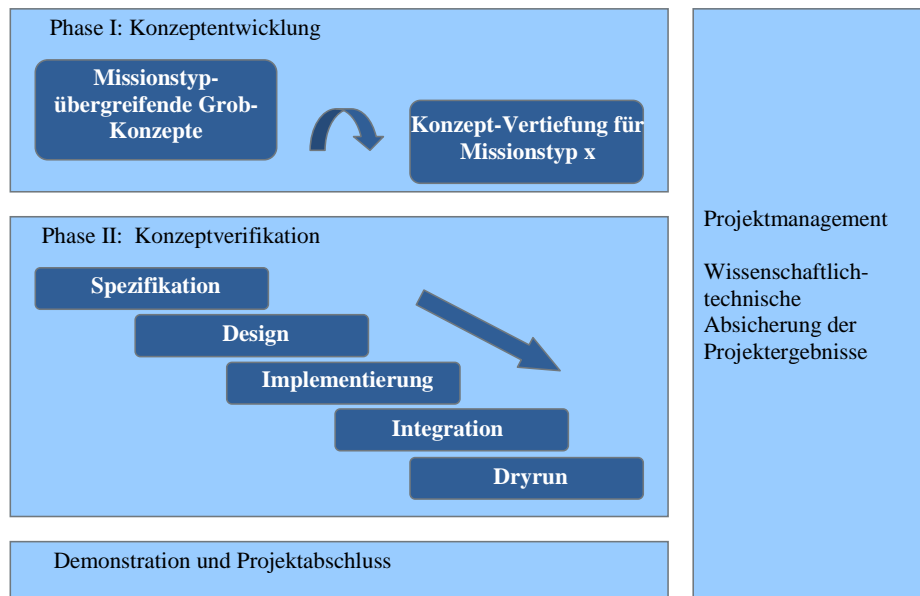


Abbildung 4: Strukturierung des Vorhabens

Der Fokus dieses Berichts liegt darauf das Gesamtergebnis des Projektes zu präsentieren.

2.4 Wissenschaftliche und technische Vorarbeiten

Für das durchgeführte Vorhaben wurden Vorarbeiten aus unterschiedlichen Projekten verwendet. Dies ist durch die komplexe Systemarchitektur begründet, die den Einsatz von unterschiedlichen Technologien einbezieht.

In den folgenden Abschnitten ist kurz beschrieben welche Vorarbeiten von den Projektpartnern eingebracht wurden.

2.4.1 Vorarbeiten der SCISYS Deutschland GmbH

SCISYS blickt auf mehr als 20 Jahre Erfahrung im Bereich Ground Systems Engineering zurück und bezieht aus einer Vielzahl dieser Projekte die benötigte Expertise im Bereich der TM/TC-Datenverarbeitung und des M&C von komplexen Bodeninfrastrukturen.

Mit der vor ca. 10 Jahren begonnenen Entwicklung des modularen Softwareframeworks egmc² konnte sich SCISYS international als Spezialist für Bodensegmentlösungen (insbesondere Ground Data Segment) etablieren. Entsprechende Systemlösungen auf Basis von egmc² sind u.a. wichtige Elemente des Columbus Control Centre, Meteosat Second Generation Programme und Galileo Control Centre.

Als aktives Mitglied der ESA Harmonisierungs-Arbeitsgruppe „Ground Systems Software“ widmet sich SCISYS seit vielen Jahren der Standardisierung und Weiterentwicklung neuer Technologien im Bereich Bodensegmente/Kontrollzentren.

Die Ergebnisse folgender Projekte wurden unmittelbar in das Projekt eingebracht:

2.4.1.1 SCOS@MAL Adapter for GSOC

SCISYS entwickelte im Auftrag DLR GSOC einen Software Adapter, um das SCOS Mission Control System mit einer Schnittstelle zu versehen, die kompatibel zu dem vergleichsweise jungen CCSDS Standard für Spacecraft Monitoring & Control (CCSDS SM&C; CCSDS-520-x,CCSDS-521-x,CCSDS-552-x) ist.

Diese Schnittstelle wurde als ein möglicher Kandidat, um den Austausch von Informationen zwischen dem Standard Mission Control System und einem separaten Kontrollsystem für die Steuerung der robotischen Komponente(n) zu ermöglichen einbezogen.

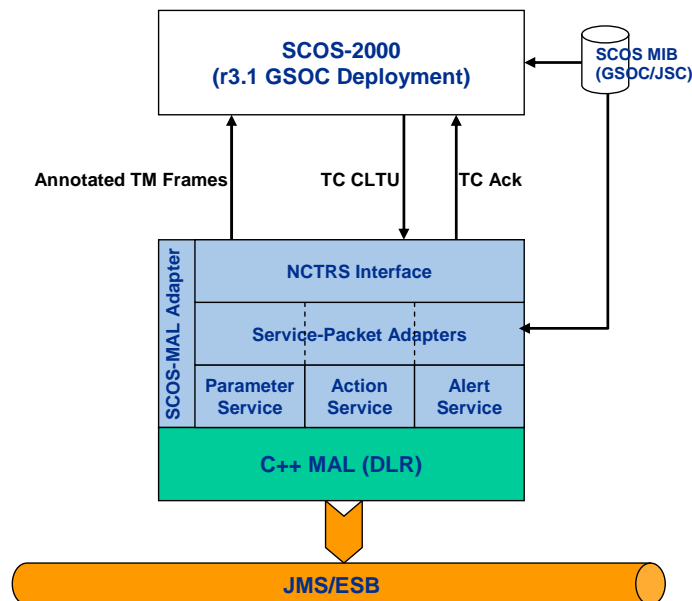


Abbildung 5: Systemarchitektur des SCOS@MAL Adapter

Neben dem exklusiv bei SCISYS vorhandenem Know-How war es der SCISYS Deutschland GmbH möglich das Expertenwissen der gesamten Unternehmensgruppe (SciSys Group) einzubringen. Relevante Arbeiten sind u.a.:

2.4.1.2 MMOPS (Mars Mission On-Board Planner and Scheduler)

Dieses Projekt beschäftigte sich mit dem Thema Intelligent Planning and Scheduling (IPS) für Rover. Ingenieure und Operator erhalten die Möglichkeit ihre Aktivitäten auf Basis von Echtzeitdaten zu planen und autonom ändern zu lassen, falls der ursprüngliche Plan nicht umsetzbar ist.

Diese zusätzliche Fähigkeit des on-board Timeline Validation, Control, and Repair (TVCR) erhöht die effektive Nutzungsdauer bzw. verlängert das Zeitfenster für wissenschaftliche Untersuchungen vor Ort.

Das Potenzial von IPS für Rovermissionen wurde erstmalig in einer repräsentativen Testumgebung demonstriert. SciSys arbeitete gemeinsam mit der University of Strathclyde (UK) und Heriot-Watt University (UK) an der Entwicklung einer on-board IPS Anwendung die in der Lage ist autonom Missionspläne auf Durchführbarkeit zu überprüfen und gegebenenfalls anzupassen.

In praktischen Versuchen wurden unter Einbindung von Fachleuten aus dem Bereich Engineering und Betrieb Vorteile und Abwägungen der Aspekte Autonomie, Komplexität, Risiken und Missionsnutzendurch durchgeführt.

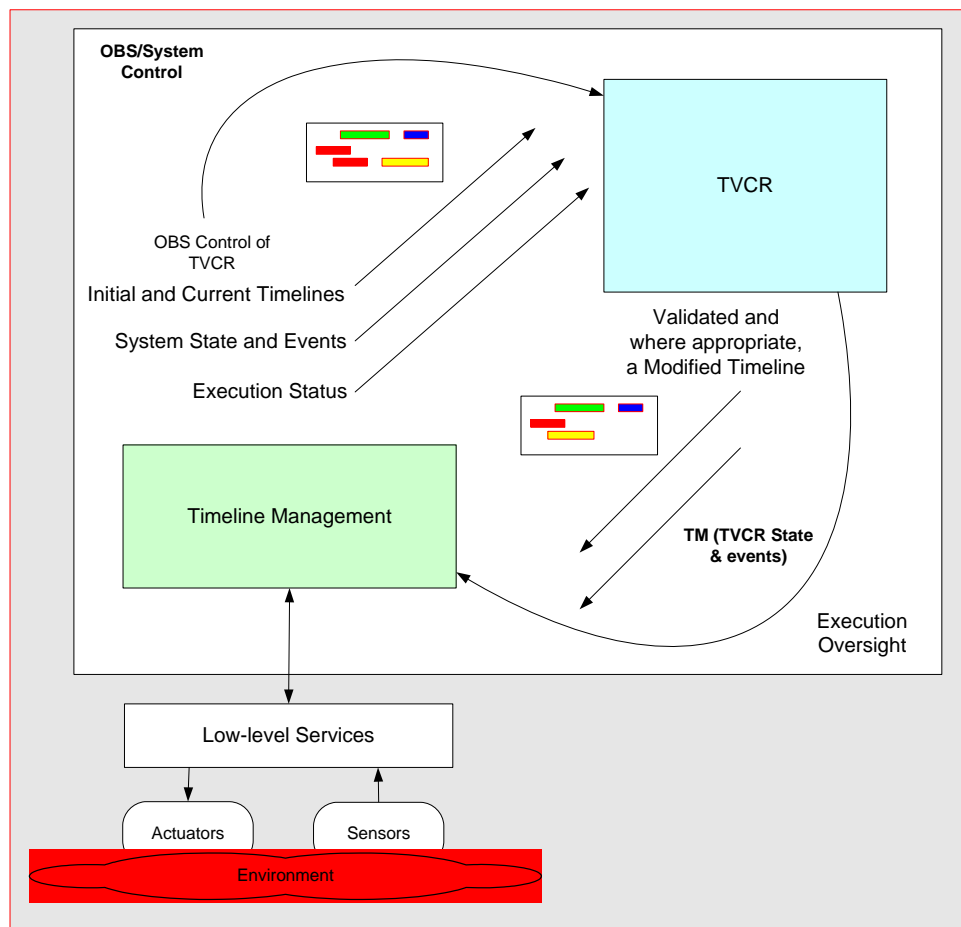


Abbildung 6: Systemarchitektur des MMOPS (Mars Mission On-Board Planner and Scheduler)

2.4.1.3 CREST

Die Projektinhalte und -ziele des Projektes waren:

- Festlegung einer initialen wissenschaftlichen Methode zur Automatisierung wissenschaftlicher Auswertungen und Planungen auf Basis terrestrischer Erfahrungswerte
- Konzept für eine Systemarchitektur zur Unterstützung autonomer opportunistischer Wissenschaft (durch Rover-Plattformen)
- Konzeptmodellentwicklung zum Nachweis der Machbarkeit und Präsentation des Modells in einem „Mars-Terrain“.



Abbildung 7: CREST Experimentaufbau

Für weitere Informationen siehe <http://www.scisys.co.uk/casestudies/space/crest.aspx>.

2.4.2 Vorarbeiten des DLR-RMC

Der Beitrag des DLR-RMC konzentrierte sich auf die speziellen OOS-Missionsaspekte, in welchem umfangreiche Vorarbeiten eingebracht werden konnten.

Als ein wichtiger Meilenstein in diesem Bereich kann das GETEX Experiment (1999) angesehen werden, bei dem die theoretischen Annahmen, die die Wechselwirkungen zwischen Manipulator und Satelliten auf einem freifliegenden System beschreiben, verifiziert wurden. Dieses Experiment wurde auf dem japanischen Experimentalsatelliten ETS-7 unter Verwendung der von der JAXA (früher NASDA) zur Verfügung gestellten Infrastruktur durchgeführt, d.h. es kamen auch hier, wie bei ROTEX, weder Planungs- noch ausgeklügelte M&C-Komponenten zum Einsatz.

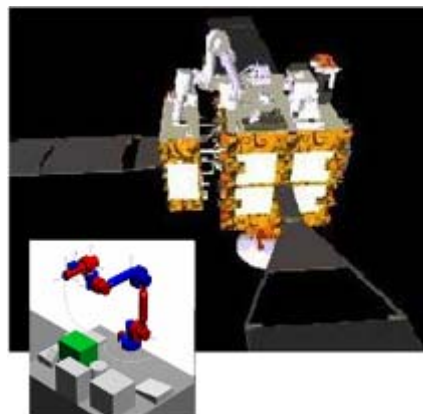
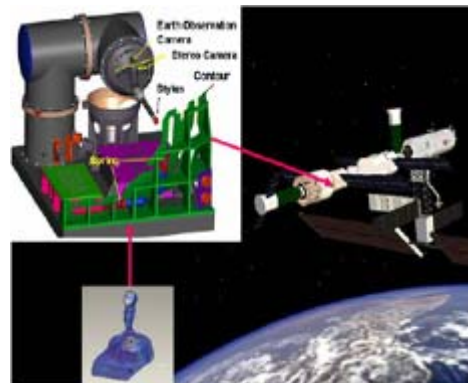


Abbildung 8: GETEX-Experiment an Bord des ETS-7-Satelliten

Das ROKVISS-Experiment (ROBotik Komponenten Verifikation auf der ISS), 2005 montiert am Svezda-Modul der ISS, diente vor allem der Verifikation der am DLR Institut entwickelten Robotergelenke für den Weltraumeinsatz. Des Weiteren wurden neue Fernsteuerungskonzepte (Telepräsenz) getestet, die für weitere geplante OOS-Vorhaben wie DEOS unbedingte Voraussetzung sind. Auch hier wurden weder Planungskomponenten entwickelt noch wurde Wert auf eine integrierte M&C Lösung gelegt.

**Abbildung 9: ROKVISS-Manipulator**

Des Weiteren sind im Raumfahrtrobotikbereich mehrere Steuerungsarchitekturkonzepte aufgesetzt worden: Das NASA/NBS Standard Reference Model for Telerobot Control System Architecture (NASREM) wurde in den 80er Jahren für die NASA entwickelt (James S. Albus, Harry G. McCain and Ronald Lumia, NASA/NBS Standard Reference Model for telerobot control system architecture, U.S. Dept. of Commerce, National Bureau of Standards, Gaithersburg, MD, 1987), um als Softwareentwicklungsrichtlinie für den Bau des Flight Telerobot Servicer (Teil des Freedom Space Station Projekts) zu fungieren, welcher aber nie gebaut wurde.

Die ESA entwickelte Anfang der 90er Jahre ein Functional Reference Model (FRM), das bereits abstrakte Planungs- sowie FDIR-Methoden beinhaltet (A. Elfving, U. Kirchhoff, Design Methodology for Space Automation and Robotics Systems, ESA Journal 1991, Vol. 15).

Auf nationaler Ebene wurde Mitte der 90er Jahre das MARCO-Projekt durchgeführt (B. Brunner, K. Landzettel, G. Schreiber, B.M. Steinmetz, G. Hirzinger, A Universal Task-Level Ground Control and Programming System for Space Robot Applications - The MARCO Concept and its Application to the ETS-VII Project i-SAIRAS 5th International Symposium on Artificial Intelligence, Robotics and Automation in Space, ESTEC, Noordwijk, The Netherlands, 1-3 June, 1999). MARCO (Modular A&R Controller) steht somit für die nationalen Anfänge im OOS Bereich und beschreibt eine generische Steuerungsarchitektur für Servicing-Satelliten. Das in MARCO vorgeschlagene und vorimplementierte Konzept wird derzeit vom DLR Institut für Robotik und Mechatronik weiterentwickelt und wurde in das Vorhaben zu wesentlichen Teilen integriert, um eine robotische Nutzlast zu simulieren.

2.4.3 Vorarbeiten des DLR-GSOC

Die Beiträge des DLR-GSOC beziehen sich auf Vorarbeiten im Bereich On-Orbit Servicing (OOS). Bei OOS Missionen ist immer der simultane Betrieb von zwei Raumfahrzeugen durchzuführen, sei es in zwei getrennten Kontrollzentren oder mit einem gemeinsamen Team.

Durch den Betrieb der Mission GRACE 1 & GRACE 2 – zweier Erdbeobachtungssatelliten, die im Formationsflug betrieben werden – verfügt das DLR-GSOC bereits seit 2002 über Erfahrung im simultanen Betrieb zweier Satelliten. Dazu gehört ein Flugbetriebssystem, welches in der Lage ist, beide Satelliten abwechselnd zu kommandieren. Die Entwicklung des Flugbetriebssystems für Tandem-X (Start Juni 2010) stellte eine weitere Herausforderung dar, da TanDEM-X und TerraSAR-X in einer sehr engen Formation mit Abständen von ca. 200 m kontrolliert werden.

Der nächste logische Schritt, die Anflugnavigation bzw. das Rendezvous & Docking (RvD) zweier Raumfahrzeuge wurde während der letzten drei Jahre im Rahmen von Phase-0, Phase-A und Phase-B Studien für die Missionen DEOS und OLEV analysiert. Insbesondere wurden dabei auch eine Reihe anderer technischer Herausforderungen wie Telepräsenz oder die Kontinuität der Kommunikationsverbindungen untersucht.

Schließlich hat das DLR-GSOC zusammen mit DLR-RMC eine Anlage zum Testen des Rendezvous und Dockings von Raumfahrzeugen entwickelt, das „European Proximity Operations System“ (EPOS). Die Testanlage EPOS wurde im Juni 2009 in einer ersten Stufe fertig gestellt und ist damit in der Lage, Sensoren und Algorithmen für die Anflugnavigation zu verifizieren.



Abbildung 10: Die EPOS-Anlage am DLR-GSOC (Stand 2009)

Neben dieser Erfahrung bring das DLR-GSOC zusätzlich die Erfahrungen für den Betrieb von Raumfahrtmissionen ein. Hierzu zählen organisatorische Aspekte wie auch die Einbringung von entsprechender Infrastruktur und Komponenten, wie etwa den Tools SATMon und SATViz zur Aufbereitung und Visualisierung von Telemetriedaten.

2.5 Zusammenarbeit mit anderen Stellen

Neben der Zusammenarbeit mit den Projektpartnern wurde der Kontakt zu anderen Industriepartnern auf entsprechenden Konferenzen gesucht und auch über die Mitarbeiter der SCISYS Deutschland GmbH zum ESOC, um zu analysieren wie die Organisationsstrukturen dort implementiert sind.

Weitere Zusammenarbeiten mit anderen Stellen haben sich innerhalb des Vorhabens nicht ergeben.

3 Projektergebnisse

3.1 Erzielte Ergebnisse

Innerhalb des Vorhabens wurde von den Projektpartnern SCISYS Deutschland GmbH (vormals VCS AG), DLR-RMC und DLR-GSOC eine Analyse der Betriebskonzepte vergangener und zukünftiger Missionen im Hinblick auf Gemeinsamkeiten durchgeführt.

Hierbei wurde in der ersten Projektphase ein repräsentatives Missionskontrollkonzept für robotische Missionen erarbeitet, welches die bestehenden Betriebskonzepte, Verantwortlichkeiten, Informationsflüsse und Missionsphasen einbezieht, bzw. berücksichtigt. Diese Ergebnisse wurden in mehreren Veröffentlichungen verarbeitet, die auf den Arbeiten des Projekts MICCRO basierten (siehe Abschnitt 3.4). neben diesen Veröffentlichungen wurden die konzeptionellen Ergebnisse in Dokumenten nach ECSS-E-ST-40C, technischen Notizen und im Rahmen des Berichtswesens festgehalten:

- Technical Notes zur Dokumentation der konzeptionellen Ideen
 - TaD: Terms and Definitions
 - TN4: Mission Operations Concept
 - TN5: Autonomy Concept
 - TN6: User Concept
 - TN18: Communication Concept
- Dokumentation nach ECSS
 - Software Specification Document (SSD)
 - Detailed Design Document (DDD)
 - Software Verification & Validation Plan (SVVP)
 - Software Transfer Document (STD)
 - Software User Manual (SUM)
- Berichtswesen
 - Zwischenberichte
 - Schlussbericht

Neben der konzeptuellen Analyse wurde ein Demonstrationsprototyp implementiert, der eine Überprüfung des entwickelten Missionsbetriebskonzepts an einem repräsentativen System incl. eines simulierten Raumsegmentes zulässt. Das Design und der aktuelle Integrationsstand des am DLR Standort Oberpfaffenhofen implementierten Systemdemonstrators wurden in einem Vortrag erläutert.

Durch die Verwendung dieses Systems konnten Aspekte wie die Übergabe zwischen verschiedenen Missionsphasen und deren Auswirkungen auf Rollen und Verantwortlichkeiten untersucht werden. Besonders für den Fall der telepräsenten Steuerung des Roboters werden neue Anforderungen in Bezug auf die Kommunikationsarchitektur aber auch die Prozesse zur Entscheidungsfindung gestellt. Die Projektergebnisse liefern hier entsprechende Lösungsansätze für zukünftige Missionen.

Ein Schwerpunkt bei der Implementierung des Demonstrationsaufbaus lag auf der Entwicklung von Komponenten wie einem integrierten Missionskontrollsystem für die Satellitenplattform und die robotische Nutzlast, Assistenzsystemen für den Operator und neuen Kommunikationskomponenten.

3.1.1 Missionsbetriebskonzept für robotische Missionen

3.1.1.1 Organisationsstrukturen

In der ersten Studienphase wurde innerhalb der MICCRO-Teams untersucht, ob das Konzept und die Organisationsstrukturen sowie die Rollen und Verantwortlichkeiten, wie sie in anderen Raumfahrtmissionen angewendet werden, auch auf robotische Missionen übertragen werden können.

Eine intuitive und bewährte Methode für das Missionsdesign und die Missionsvorbereitung ist es die Struktur der primären Raumsegmentfunktionen und –schnittstellen auf das Bodensegment zu übertragen. Hierdurch sollte die Bodensegmentinfrastruktur im Hinblick auf die Organisation aber auch die Anzahl und Komplexität der Schnittstellen zwischen verschiedenen Komponenten bereits minimiert sein.

Ein Beispiel für die Einführung einer wohldefinierten Schnittstelle im Bodensegment ist eine existierende HF-Verbindung zwischen korrespondierenden Flugkomponenten. Hier kann z. B. für eine Mars-Explorationsmission die Schnittstelle zwischen einem Orbiter und einem Rover auf der Planetenoberfläche genannt werden. Beide Komponenten sind nur lose gekoppelt. Mit Blick auf die Bodeninfrastruktur ist somit auch der Betrieb aus zwei unterschiedlichen Kontrollzentren möglich und sinnvoll.

Für OOS-Missionen wie DEOS hingegen besteht eine sehr starke und direkte Kopplung zwischen der robotischen Komponente, z. B. einem Manipulator und der Satellitenplattform, da eine Bewegung des Manipulators immer auch auf die Plattform zurückkoppelt und so die Lage des Gesamtsystems beeinflusst. Hierdurch wird auch innerhalb der Flugkomponente eine enge Interaktion zwischen der robotischen Komponente und des AOCs erforderlich. Weiter wird für einen Zugriff die Kollisionsvermeidung zu einer Herausforderung, da sowohl der Roboter- als auch der Plattformbetrieb einen direkten Einfluss haben. Für diesen Missionstyp ist es deshalb von entscheidender Bedeutung, den Betrieb der Satellitenplattform und der robotischen Komponente parallel aus einem Kontrollraum mit einem integrierten Betriebsteam durchzuführen.

Ein integriertes Flight Operations Team (FOT) hat hierbei speziell im Hinblick auf den Überblick aller Komponenten und eine kurzfristige Entscheidungsfindung große Vorteile. Die hierarchische Befehlsstruktur, wie sie in den Kontrollzentren etabliert ist, passt ebenfalls gut auf die Gesamtstruktur. Für den Telepräsenzbetrieb und den Betrieb von Echtzeitsystemen hingegen müssen diese Befehlsstrukturen entsprechend angepasst werden.

Es wird eine direkte Kontrollschleife parallel zur bestehenden TM/TC-Verbindung benötigt, um eine Antwortzeit zwischen Roboter auf dem Satelliten und dem Robotic-Operator innerhalb von maximal 100ms zu ermöglichen.

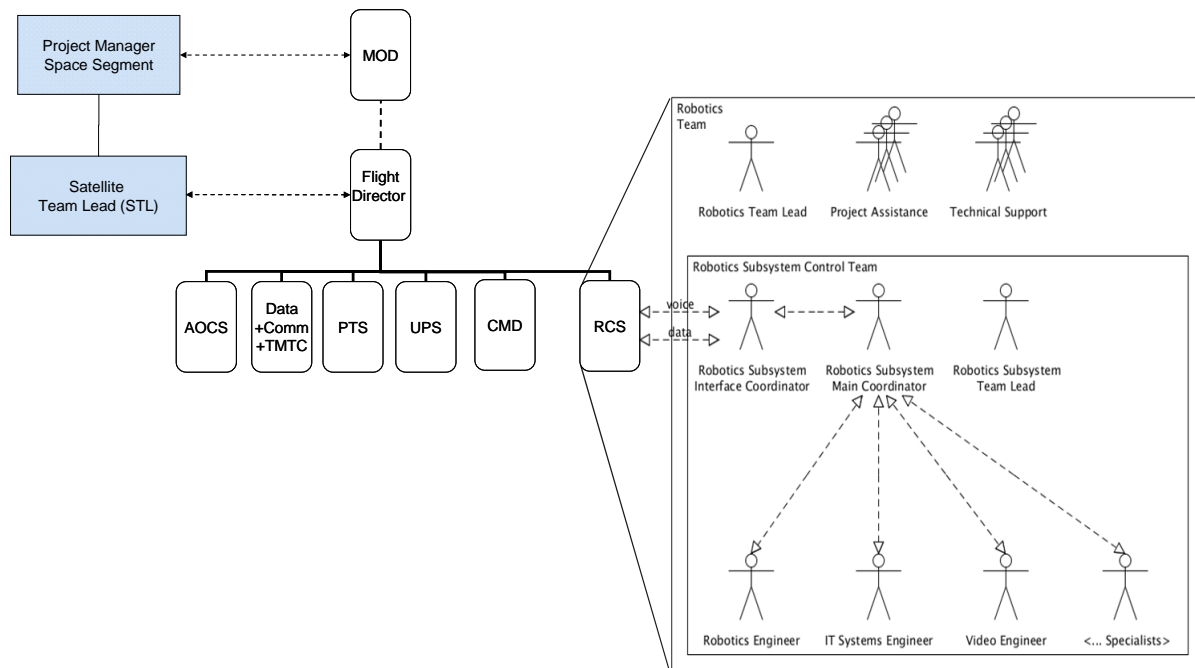


Abbildung 11: Organisationsstruktur im Kontrollraum

3.1.1.2 Rollen und Verantwortlichkeiten

Um die Zuordnung der Verantwortlichkeiten zwischen Flugdirektor (FD) und dem Robotik-Operator im Detail zu beschreiben, wurde eine Analyse anhand von grundsätzlichen Kriterien für die Erstellung eines Konzeptes für Rollen und Verantwortlichkeiten mit Blick auf robotische Missionen durchgeführt.

In diesem Zusammenhang wurden die folgenden zentralen Fragen gestellt:

- Wie schnell müssen Entscheidungen getroffen werden?
- Welche Informationen sind für eine Entscheidungsfindung notwendig?
- Beinhaltet der robotische Betrieb auch Telepräsenz?
- Können bestimmte Verantwortlichkeiten unabhängig von anderen betrachtet werden?
- Wird hierdurch ein integriertes Missionsbetriebsteam in einen gemeinsamen Kontrollraum notwendig oder ist ein verteiltes Team in unterschiedlichen Kontrollräumen möglich?

Mit Blick auf die sehr kurzen Zeitskalen im Telepräsenzbetrieb erscheint eine Kommandierung mittels der Voice-Loop, wie sie sonst als Standard eingesetzt wird, ungeeignet. Die Befehlskette aus Robotik-Operator, FD und Command-Operator (CMD) muss für diesen Betriebsmodus abgeändert werden. Zusätzlich muss auch für andere Subsysteme eine direkte Steuerung seitens des Robotik-Operators ermöglicht werden (AOCS). Die Verantwortung für den Betrieb sollte hierdurch unbeeinflusst beim Flugdirektor bleiben.

Die Rollen und Verantwortlichkeiten zwischen FD und Robotik-Operator sind vergleichbar mit einem Space Shuttle Kommandanten und seinem Pilot: Während der Robotik-Operator das Kommando über das Subsystem hat und zusätzlich eingeschränkte Kontrolle über das Gesamtsystem für den Telepräsenzfall, hat der FD zu jeder Zeit die volle Verantwortung für das Gesamtsystem. Für Entscheidungen die im Zeitrahmen von Sekunden und Minuten getroffen werden, werden diese auch direkt vom FD ausgelöst.

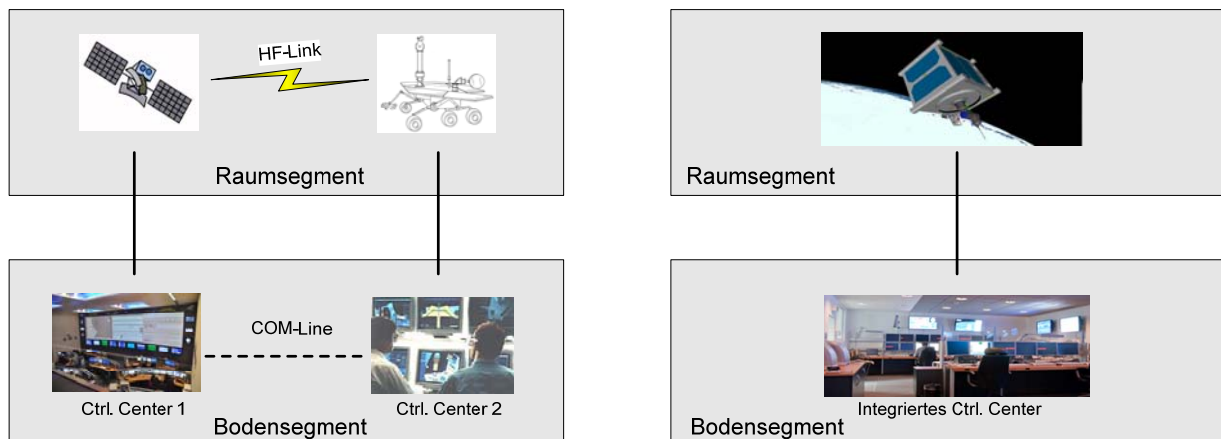


Abbildung 12: Abbildung des Raumsegmentes auf das Bodensegment

3.1.1.3 Kommunikation

Die Kommunikationsinfrastruktur ist ein wesentlicher Bestandteil aller Weltraummissionen. Einige allgemeine Randbedingungen müssen während ihres Designs beachtet werden:

- **Finanzielle Randbedingungen:** Kosten führen zu Einschränkungen bei Weltraummissionen, was besonders auf das Bodensegment mit seinem Bodenstationsnetzwerk zutrifft. Die Ausführung der Routineoperationsphase einer Weltraummission stellt daher immer einen Kompromiss zwischen einem höheren Grad an On-Board-Autonomie oder -Robustheit und der Anzahl der (teuren) Bodenstationskontakte dar, die zur Kontrolle des Raumfahrzeugs benötigt werden. Das hat natürlich Konsequenzen für das Design und den Test des Raumfahrzeugs, denn die Entwicklung eines hochgradig autonomen Satelliten ist teurer als der Bau eines „einfachen“ Raumfahrzeugs, welches häufiger vom Boden aus gesteuert werden muss.
- **Konnektivitätseinschränkungen:** Die benötigte Bandbreite, die Latenzzeit und das Protokoll müssen analysiert und werden und die Bodenstationen zur Unterstützung der Mission müssen ebenfalls den Anforderungen entsprechen.
- **Raumsegmenteinschränkungen:** Das Raumsegment muss eine Anzahl missionspezifischer Kommunikationsanforderungen erfüllen. Diese können nicht immer im Voraus beschrieben werden, da sie stark vom Kommunikationsweg und von möglichen anderen Raumfahrzeugen abhängen. Ein Beispiel für spezielle Einschränkungen ist durch die Telepräsenzanforderung gegeben, die später beschrieben wird.

Für robotische Anwendungen müssen zusätzliche Einschränkungen in Betracht gezogen werden. Teleoperation ermöglicht zeitversetzten Zugang zu einer Anwendung im Weltraum. Die Kommandos für jeden Prozessschritt müssen im Voraus identifiziert werden und anschließend müssen alle Schritte in einer skriptartigen Prozedur zusammengesetzt werden. Direkte Interaktion, besonders in Notsituationen oder Katastrophenfällen, ist in der Regel unmöglich.

Telepräsenz ermöglicht ein direktes Feedback zu einem Operator über die physische Korrelation zwischen Aktion und Reaktion. Sie ermöglicht einen flexiblen Umgang mit der aktuellen Situation, was besonders für Wartung und Reparatur von Nutzen ist. Telepräsenz wird durch einen verteilten Regelkreis realisiert und stellt daher im Vergleich zur Teleoperation strengere Forderungen an die Kommunikationsverbindung.

Eine Kommunikationsverbindung für Telepräsenz ist charakterisiert durch Bandbreite, Jitter, Latenz, Verlässlichkeit der Übertragung, Kontaktdauer und Signalübertragungszeit, und jedes dieser Kriterien hat direkten Einfluss auf die Qualität des Services und der Regelung. Wenn Telepräsenz das Feedback der On-Board-Reaktionen beinhaltet, die durch operatorinitiierte Aktionen verursacht werden, reduziert sich die maximale Operationsdistanz auf den Geostationären Orbit (GEO) und die Rundlaufzeit auf zirka 500 ms. Aktuelle Standardkommunikationsinfrastruktur wurde während der

Studie analysiert um eine optimale Herangehensweise zur Minimierung von Jitter und Latenz zu gewährleisten.

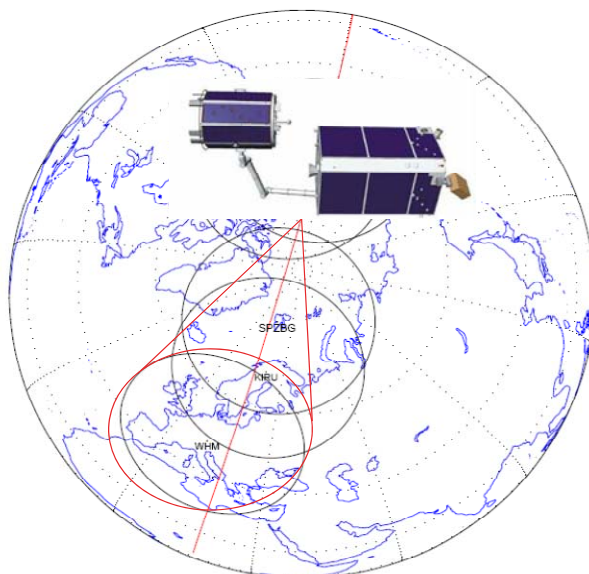


Abbildung 13: Verlängerung der Kontaktzeit durch ein Netzwerk aus Bodenstationen

Die gesamte Bodenkontaktzeit eines Satelliten je Orbit kann durch die Nutzung eines GEO Relaysatellitensystems, z.B. des National Aeronautics and Space Administration's (NASA) Tracking and Data Relay Satellite System (TDRSS), oder durch ein Netzwerk von Bodenstationen mit oder ohne überlappende Empfangsgebiete verlängert werden (vgl. Abbildung 13). In beiden Fällen steht die Telemetrie für den gesamten Kontakt lückenlos zur Verfügung. Bei einem Relaysatellitensystem gilt dies auch für die Möglichkeit, Telecommandos abzusetzen. Bei einem Bodenstationsnetzwerk ist dies aufgrund der benötigten Handover-Prozedur zwischen jeweils zwei Bodenstationen für einen kurzen Zeitraum nicht möglich. Bedingt durch die erhöhte Komplexität und die komplizierte Hardware haben beide Optionen Einfluss auf die Latenzzeit und benötigen eine gute Unterstützung durch entsprechende Planungstools, die auch auf kurzfristige Änderungen reagieren.

Der Grundgedanke, der bei allen ferngesteuerten Operationen zur Anwendung kommt, ist die Übertragung von Informationen innerhalb logischer/virtueller Kanäle, die im Multiplexverfahren über eine physikalische Funkverbindungen übertragen werden (siehe Abbildung 14). Die mögliche physikalische Verbindungsbandbreite wird dabei automatisch und prioritätsbasiert über alle aktiven Kanäle verteilt. Eine flexible Anpassung an wechselnde Anforderungen ist wichtig, denn die effektiv mögliche Verbindungsbandbreite ist umgekehrt proportional zur Signalübertragungsstrecke.

Abgesehen von der physikalischen Übertragungskapazität können die übertragenen Daten in drei Qualitätsstufen eingeteilt werden:

1. Zyklisch versendete synchrone Daten, z.B. Onlinetelemetrie
2. Azyklisch oder eventgesteuert versendete Daten, z.B. Telecommandos
3. Synchrone Daten und asynchrone Daten, auf Anfrage versendet, z.B. Housekeeping Daten, Dateitransfers

Für die Datenübertragung sind mehrere Protokollvorschläge aus verschiedenen Quellen erhältlich, z.B. CCSDS, ECSS und TCP/IP. Da diese Standards weit verbreitet sind, bilden sie die Grundlage für das Kommunikationsdesign von MICCRO.

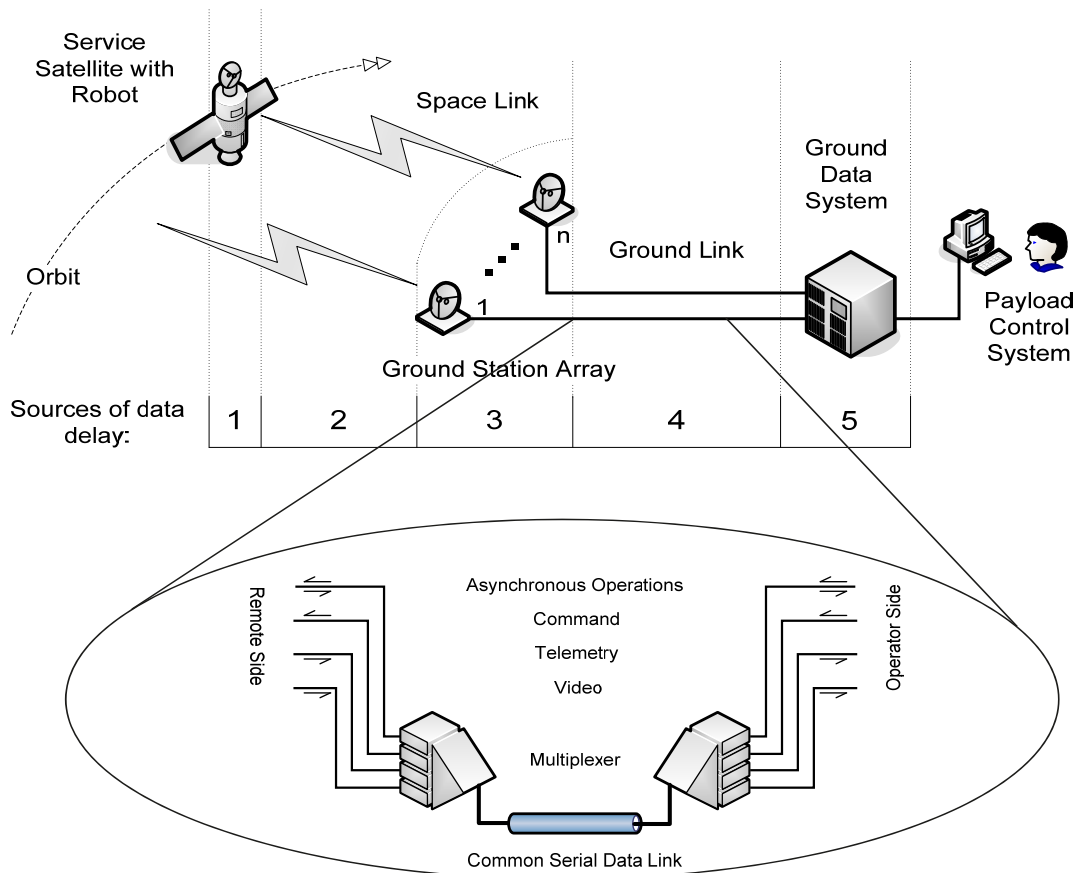


Abbildung 14: Implementierung virtueller Kommunikationskanäle

Abbildung 15 zeigt das Konzept einer Kommunikationsstruktur, das in MICRO erarbeitet wurde. Das Raumfahrzeug wird über SCOS 2000, ein Standard Mission Control System, überwacht und gesteuert. Die Anbindung geschieht wie gewohnt über ein SLE Netzwerk. Der robotische Anteil des Raumfahrzeugs wird über ein Robotic Control System bedient. Für Teleoperation ist es über die EXIF Schnittstelle an SCOSS 2000 angebunden. Für die Telepräsenz steht eine direkte Datenleitung zur Bodenstation zur Verfügung, die die hohen Anforderungen der Telepräsenz erfüllen kann. Ein Communication Gateway übernimmt das Multiplexing der Daten und gibt diese an den Cortex weiter. Ab hier werden die Daten wie gewohnt über eine Antenne zum Raumsegment transferiert.

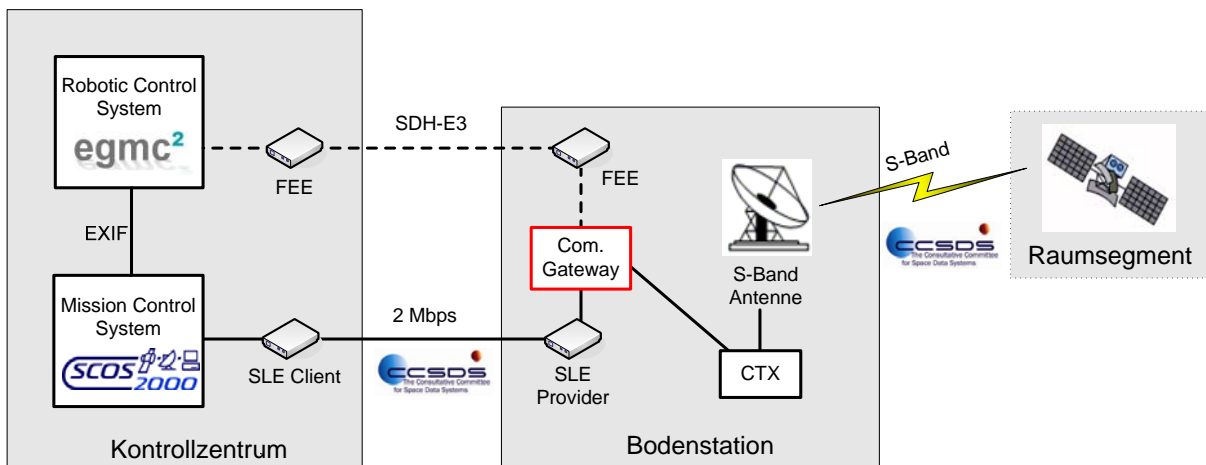


Abbildung 15: Kommunikationsstruktur

3.1.1.4 Autonomie

Autonomie hat im Hinblick auf robotische Missionen eine Vielzahl unterschiedlicher Aspekte, die im Rahmen der MICCRO-Studie intensiv analysiert wurden.

Autonome Funktionen verteilen sich über verschiedene Anwendungsdomänen von Raumfahrtmissionen, z. B. Guidance Navigation and Control (GNC); Failure Detection, Isolation and Recovery (FDIR); Managing and Intelligent Sensing and Data Handling. Hierbei werden diese Funktionen typischerweise sowohl am Boden als auch an Bord implementiert. Eine Vielzahl von Technologien wird derzeit bereits eingesetzt, um eine grundlegende Autonomie zu implementieren. Beispiele sind hier unter Anderem: Scheduler, Event-basierte Ausführungssysteme und Star Trackers mit integrierter Selbstlokalisierung. Unterschiedliche Arten der Autonomie, wie sie möglicherweise vorteilhaft in robotischen Missionen eingesetzt werden kann, wurden in MICCRO diskutiert.

Spezielle Anforderungen an die Onboard Autonomy wurden bereits für die Anflug- und Dockingphase von OOS-Missionen identifiziert. Aufgrund der begrenzten Hardwareleistungsfähigkeit an Bord, erscheint es nicht sinnvoll das komplette dynamische Modell an Bord des Servicers zu berechnen und somit die Rechenlast auf den Boden zu verlagern. Hierdurch muss eine enge Anbindung der Onboard Sensoren, Aktuatoren und den Bodenkomponenten gewährleistet sein. Bemerkenswert ist, dass durch diese Implementierung die Kontrolle implizit – speziell für Aktivitäten mit Telepräsenzsteuerung - auf den Boden verlagert wird. Somit kommen allerdings auch neue Fehlerszenarien hinzu, die durch einen Verbindungsabbruch ausgelöst werden. Eine Lösungsansatz hierfür ist ein Collision Avoidance Maneuver (CAM), das eine Rückzugsstrategie verwendet, die immer basierend auf dem aktuellen Systemstatus berechnet wird und regelmäßig für eine Ausführung an Bord hochgeladen und aktualisiert wird.

Bodenautonomiekonzepte können auf zwei unterschiedliche Arten implementiert werden:

1. Der betrieb wird direkt beeinflusst, z.B. in Form einer Procedure Execution (PEX), welche Ereignis getriebene Aufgaben als autonome Reaktion ausführt.
2. Alternativ kann einen Autonomiekomponenten bewusst entkoppelt vom realen System betrieben werden, um dem Operator so unabhängig als Assistenzsystem zur Verfügung zu stehen.

Letztere Lösung wurde in MICCRO implementiert um dem Operator als Assistenzsystem zu dienen, wobei die Kontrolle immer beim Operator bleibt.

3.1.1.5 Benutzerschnittstelle

Benutzerschnittstellen (Human Maschine Interfaces, HMI) erlauben Operatoren nicht nur Informationen über den aktuellen Zustand eines Systems zu erhalten, sondern auch, falls nötig, den Systemzustand zu ändern. Für eine robotische Raumfahrtmission stellen die Monitoring & Control (M&C) Software Suite SCOS-2000 und das SCISYS egmc² Framework bewährte HMIs bereit, die bereits erfolgreich eingesetzt wurden, um andere Weltraummissionen zu unterstützen. Das neue robotische Missionselement benötigt jedoch eine weitere HMI-Komponente, die im Rahmen von MICCRO implementiert wurde.

Im Allgemeinen gibt es drei Kategorien von Daten, die menschlichen Operatoren durch HMIs präsentiert werden müssen:

1. Satelliten Uplink/Downlink Daten, die sowohl konventionelle Kommando- und Telemetriedaten enthalten, als auch Daten, die durch Echtzeitkanäle übertragen werden
2. Daten, die zwischen Operatoren und anderen Anwendungen auf dem Boden ausgetauscht werden, wie z.B. Bodendaten, Bodenautonomie, Videoverarbeitung und ähnliche Hilfsanwendungen.
3. Management- und Statusdaten für alle Bodeneinrichtungen, einschließlich Mittel zum Starten, Stoppen, Zurücksetzen und Überwachen dieser Systeme

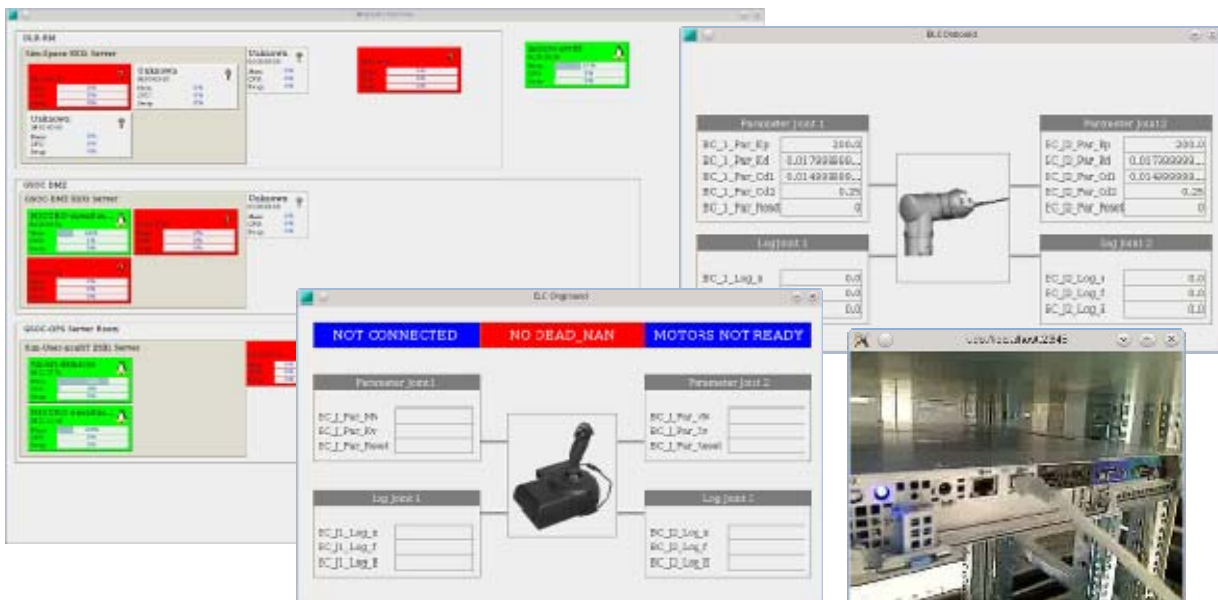
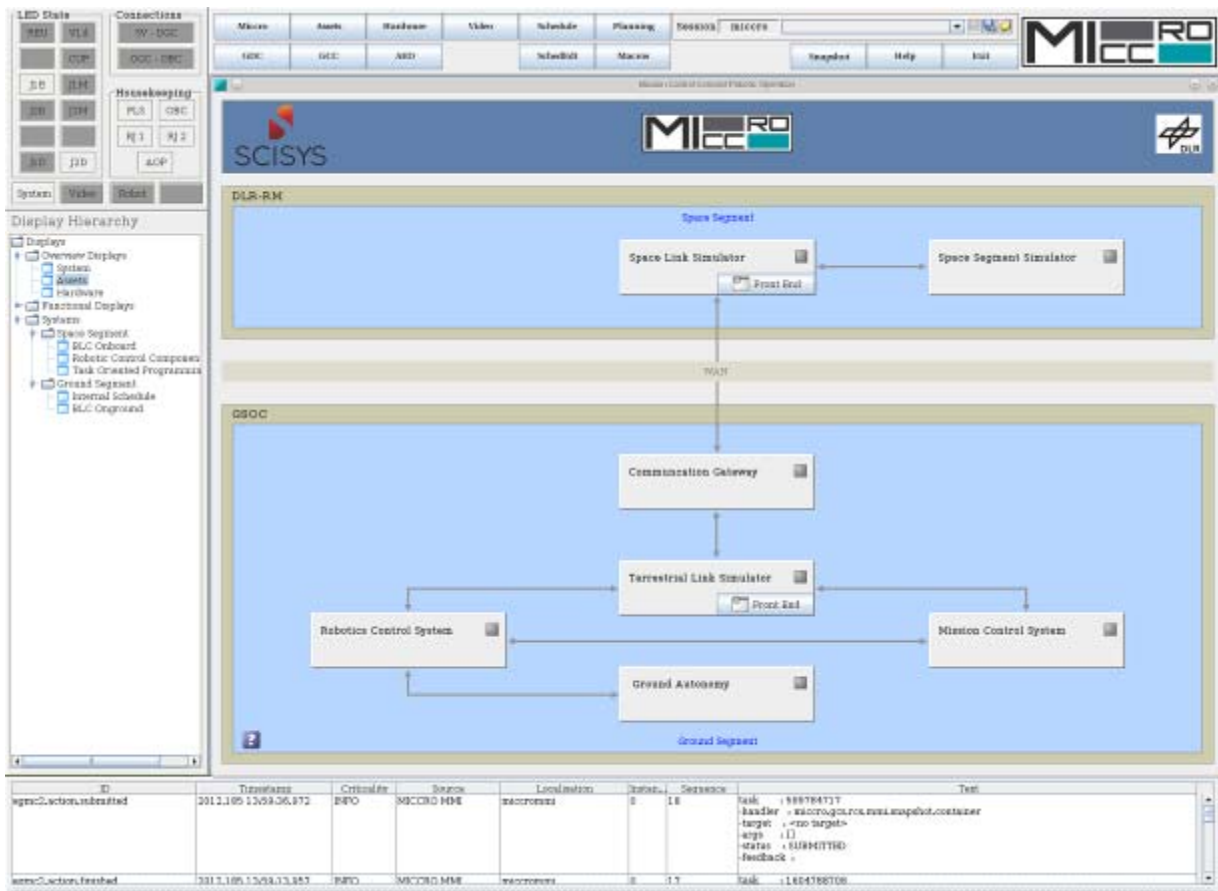


Abbildung 16: Benutzerschnittstelle

Diese Daten sollten durch HMI-Komponenten, welche in ein standardisiertes Anwendungsframework passen, visualisiert werden (vgl. Abbildung 16). Die Zusammensetzung dieser Komponenten variiert hinsichtlich der Verantwortlichkeiten des jeweiligen Operators, wobei manche Komponenten wiederholt für mehrere oder alle Schnittstellen verwendet werden können. Aus Erfahrungen, die in früheren Missionen gesammelt wurden, konnten folgende Anforderungen aus robotischer Sicht zusammengestellt werden:

- Die Nutzerschnittstellen für die Roboteroperatoren sollten high-level Softwareergonomie-Standards entsprechen
- Aus demselben Grund sollte dem Roboteroperator eine Augmented-Reality-Darstellung der Situation an Bord des Raumfahrzeugs zur Verfügung gestellt werden
- Die HMI-Komponenten sollten eine Zusammenfassung der Zustände von Gesamtsystemen am Boden und an Bord beinhalten, sowie eine detailliertere Zusammenfassung der robotischen Subsysteme liefern.
- Ein Missionszeitplan sollte geplante und tatsächliche Prozeduren und Ereignisse, sowie extern getriggerte Ereignisse, wie Finsternisphasen, wiedergeben.
- Eine Videodarstellung sollte eine Vorstellung der Situation an Bord für alle Roboteroperatoren erleichtern.
- High-end haptische Eingabegeräte sollten den Roboteroperatoren die Möglichkeit geben, tatsächlich zu fühlen, was an Bord während der telepräsenen Manipulation geschieht.

3.1.2 Systemdemonstrator

Dieser Abschnitt beschreibt den Systemdemonstrator in seinen wesentlichen Komponenten, wie er am 20.09.2012 öffentlich vorgeführt wurde. Ein Blockschaltbild der wesentlichen Funktionsblöcke ist in der folgenden Abbildung gezeigt:

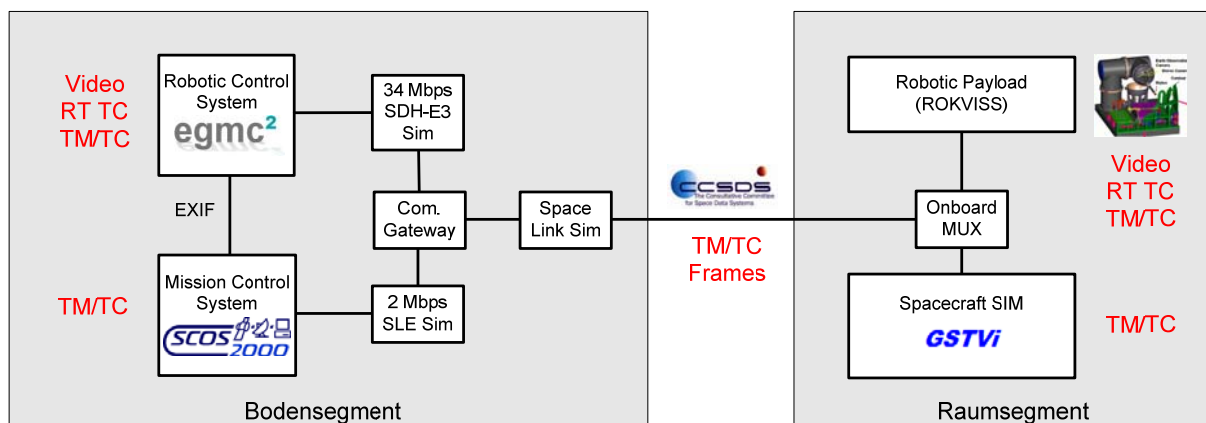


Abbildung 17: End-to-End-Systemdemonstrator

Der Systemdemonstrator ist in Boden- und Raumsegmentkomponenten unterteilt. Hierbei wurde für die Demonstration das Raumsegment durch entsprechende Simulatoren und Emulatoren repräsentiert.

Das Bodensegment wird durch das Mission Control System (MCS) zur Kontrolle und Überwachung der Satellitenplattform genutzt. Über dieses System werden sowohl die Telemetrie (TM) als auch die Telekommandos (TC) für die Satellitenplattform aber auch die robotische Nutzlast empfangen bzw. gesendet. Für die Steuerung der robotischen Nutzlast kommt das Robotic Control System (RCS) zum Einsatz. Die Datenströme aus diesen beiden Subsystemen werden über Linksimulatoren, die die Übertragungskanalcharakteristiken abbilden können, geführt, um dann im Communication Gateway sämtliche Daten zum Raumsegment zu multiplexen bzw. demultiplexen.

Für das Raumsegment wurde für die Darstellung der robotischen Nutzlast das Engineeringmodell aus der ROKVISS-Mission verwendet, um eine möglichst realistische Darstellung zu erreichen. Die unterschiedlichen virtuellen Kanäle wurden analog zum Communication Gateway am Boden durch den Onboard MUX verwaltet. Die Telemetrie und Telekommandos an die Satellitenplattform wurden durch den Spacecraft Simulator GSTVi der ESA emuliert. Hierbei emuliert GSTVi zusammen mit dem als Plugin realisierten Rfz-Simulators (inkl. rudimentärer Bahndynamik) das Bodensegment inkl. Antenne sowie das Raumsegment.

Für die Implementierung des Gesamtsystems wurde weitest möglich auf Standardkomponenten zurückgegriffen. Dies konnte jedoch nicht für alle Funktionsblöcke erreicht werden, so dass auch neue

Softwarekomponenten implementiert wurden, um auf die speziellen Anforderungen für robotische und speziell OOS-Missionen einzugehen. Hier wurden z.B. die Komponenten zum Multiplexen und Demultiplexen der unterschiedlichen virtuellen Datenkanäle in Form des Communication Gateways und des Onboard MUX realisiert.

Die Systemarchitektur des Demonstrators ist als Weiterentwicklung der Architektur des ROKVISS Experiments zu verstehen. Da in ROKVISS keine Satellitenplattform zu steuern war, wurde kein vollständig ausgeprägtes MCS benötigt. Stattdessen wurden die benötigten Telecommandos und die empfangene Telemetrie über eine eigens für das Experiment entwickelte Software erzeugt und versendet bzw. dargestellt. In zukünftigen OOS-Missionen ist jedoch die Integration eines MCS wie SCOS 2000 unabdingbar. Das Communication Gateway ermöglicht nun diese Integration und bietet dabei gleichzeitig einen echtzeitfähigen Telepräsenz-Kanal wie bei ROKVISS.

Für die Installation in der letzten Integrationsstufe des MICCRO Demonstrators wurde die Mehrheit aller Hardware- und Softwarekomponenten im GSOC in die bestehenden Infrastrukturen integriert. Hierzu wurden umfangreiche Vorbereitungen durchgeführt, die sich auf die Bereitstellung der Serverhardware, Kontrollraumkapazität und auch die Konfiguration der Netzwerkinfrastruktur bezogen.

Für die Installation der virtuellen Plattformen konnten GSOC Plattformen verwendet werden, die in den unterschiedlichen Subnetzwerken angebunden sind. Hiermit einhergehend musste sicher gestellt werden, dass alle Router und Firewalls den entsprechenden Zugriff der Plattformen untereinander aber nach außen zu den im RMC installierten Komponenten erlauben. Der Aufbau der Netzwerkinfrastruktur ist in nachfolgender Abbildung dargestellt.

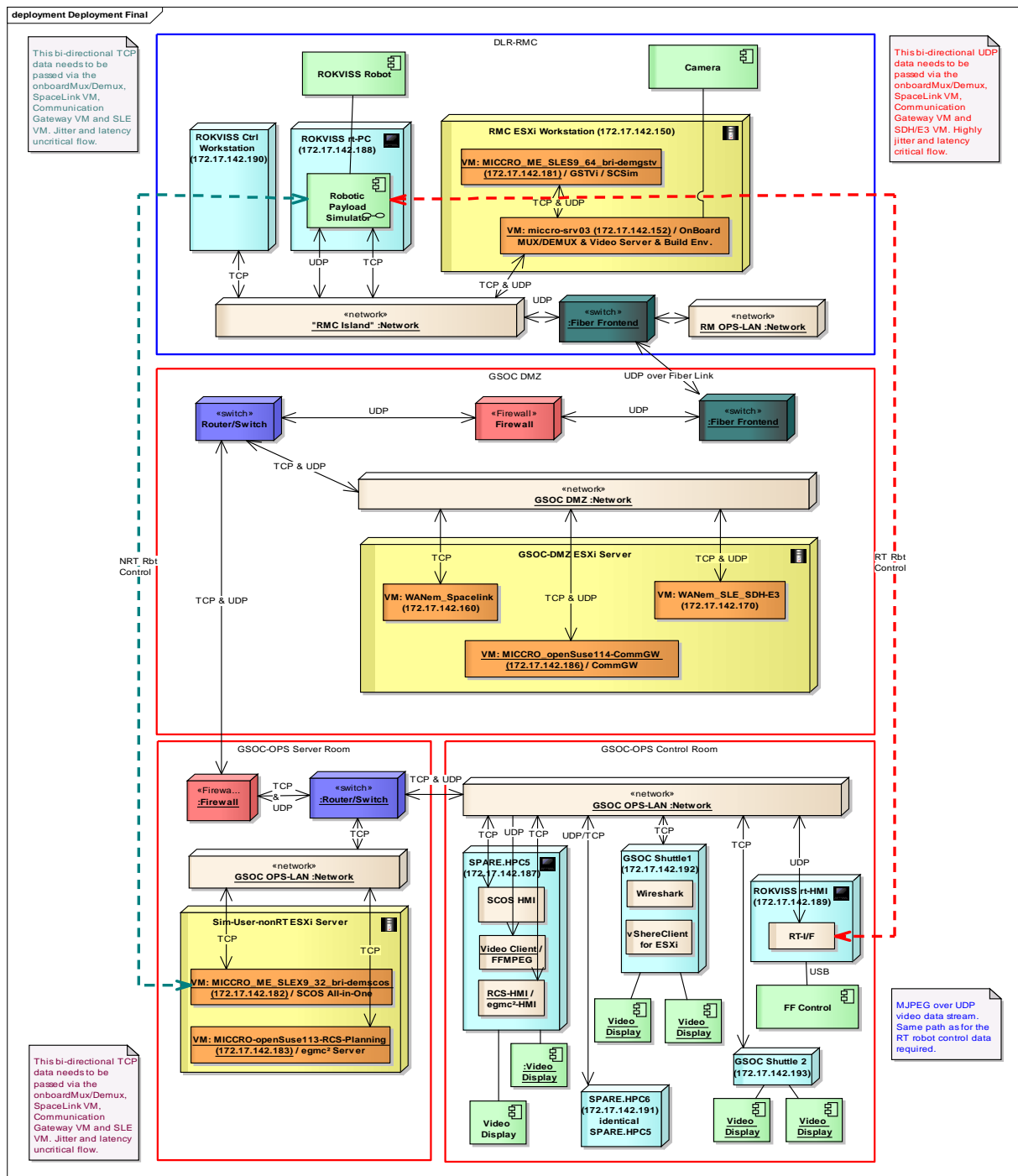


Abbildung 18: Hardwarekomponenten des Systemdemonstrators

Die Integration in die Netzwerktopologie des GSOC, sowie die Integration der Komponenten außerhalb des GSOC stellte sich als sehr herausfordernd heraus, zumal die Realtime-Anforderungen berücksichtigt werden mussten. Abbildung 19 zeigt schematisch die Netzwerktopologie. Die Systemkomponenten sind über drei Subnetze verteilt installiert und ein Datentransfer erfolgt über eine hohe Anzahl von Routern und insgesamt drei Firewalls. Auch unter diesen Voraussetzungen konnte eine Round Trip Time von ca. 10 ms erzielt werden. Diese sehr geringe Latenz ist somit ein guter Ausgangspunkt für die weiteren Untersuchungen in Bezug auf den Einfluss der Gesamtlatenz und des Jitters wie sie über die Linksimulatoren induziert werden können.

Für den subnetzwerkübergreifenden Zugriff auf die einzelnen Plattformen wurde eine entsprechende Adressübersetzungsregeln (Network Address Translation, NAT) konfiguriert, um eine gegenseitige Erreichbarkeit der Plattformen innerhalb und außerhalb des GSOC zu ermöglichen.

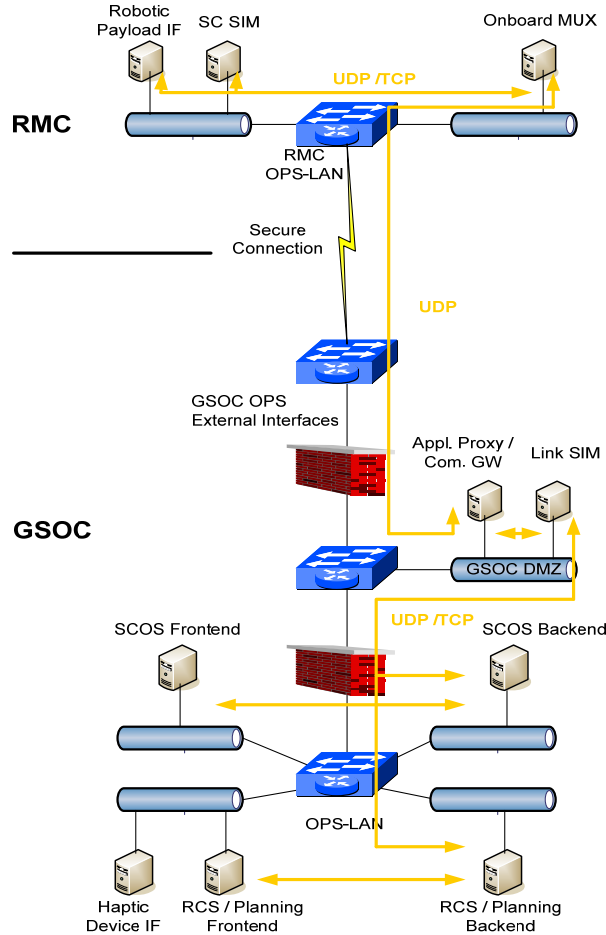


Abbildung 19: Aufbau der Netzwerkinfrastruktur und der installierten Plattformen des MICCRO Demonstrators

Implementiert wurde das System in einer verteilten Hardwareumgebung, die auch eine räumliche Trennung des Boden- und Raumsegments beinhaltet. Dies ist auch in der folgenden Abbildung veranschaulicht.



Abbildung 20: Räumliche Aufteilung der Systemkomponenten

Der in Abbildung 20 auf der rechten Seite gezeigte Kontrollraum wurde als Kontrollraum für mehrere Missionen verwendet (Multi Mission Room). Die Anordnung der einzelnen Konsolen und Operator-Arbeitsplätze ist in der nachfolgenden Abbildung gezeigt:

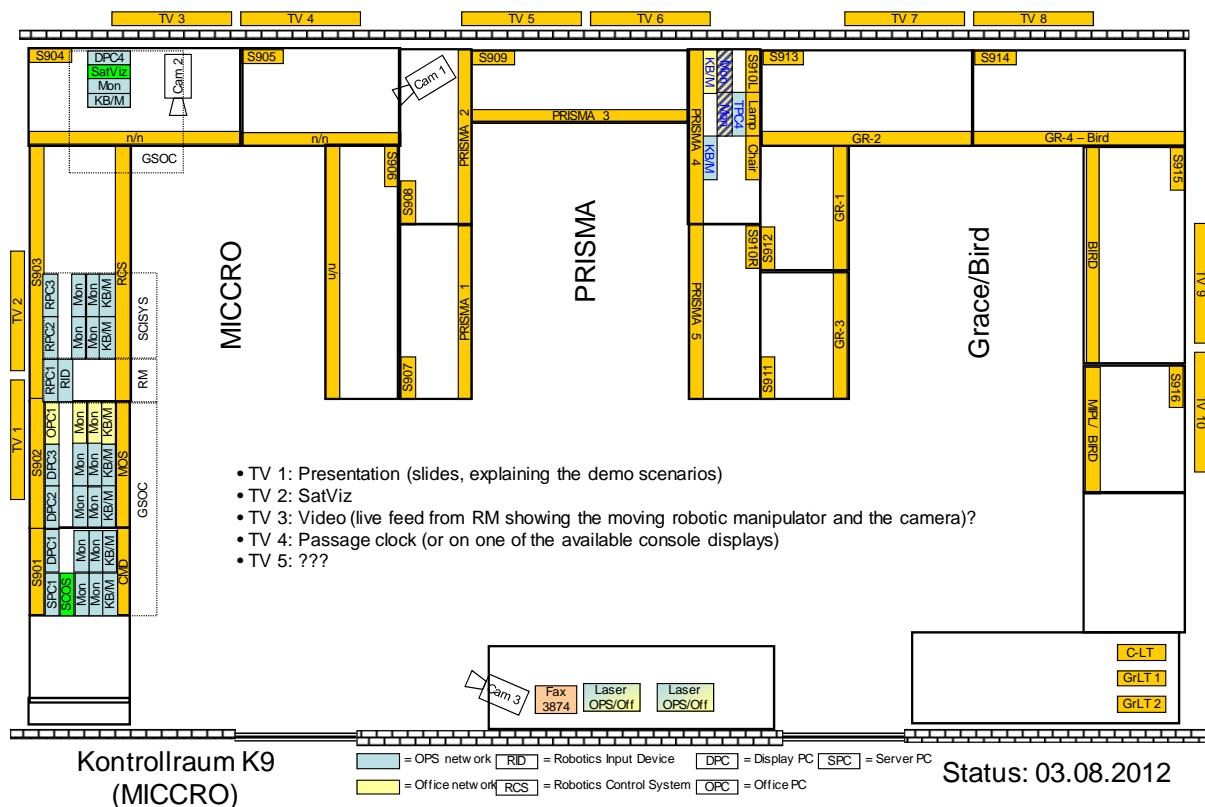


Abbildung 21: Konsolenzuordnung und Raumaufteilung im Kontrollraum

3.1.2.1 Robotic Control System (RCS)

Für die Steuerung der robotischen Komponenten wurden verschiedene Subsysteme zu einem einheitlichen Robotic Control System (RCS) integriert. Dieses besteht im Wesentlichen aus den folgenden vier Komponenten:

- RCS/Planning Server: Dieser zentrale Serverbasiert auf dem egmc²-Framework, und verwaltet die Telemetrie und Kommandos der robotischen Komponente. Zusätzlich dient der Server zur Orchestrierung der Planungsprozesse für die Ground Autonomy Komponente sowie als Verbindungspunkt für die HMIs des RCS. Schnittstellen sowohl zu SCOS (EXIF) als auch zur Telepräsenz-Steuerung wurden implementiert. Innerhalb des RCS Server wurde eine Überwachung der Hardwarekomponenten realisiert.
- Benutzerschnittstelle: Die Benutzerschnittstelle basiert auf dem egmc²-Framework und integriert auch externe Anwendungen wie etwa zur Darstellung des für die Telepräsenz-Steuerung notwendigen Videodatenstroms.
- Telepräsenz-Steuerung: Die telepräzente Steuerung des Roboters erfolgt wie bei ROKVISS über einen dedizierten Echtzeit-Datenkanal. Die Überwachung der On-ground TM und die Konfiguration der Telepräsenzparameter können nun direkt aus dem RCS Server über die bereitgestellte MCC-API durchgeführt werden.
- Hardware: Für die Steuerung des Roboters ist ein Joystick erforderlich. Dieser wird ebenfalls dem RCS zugeordnet.

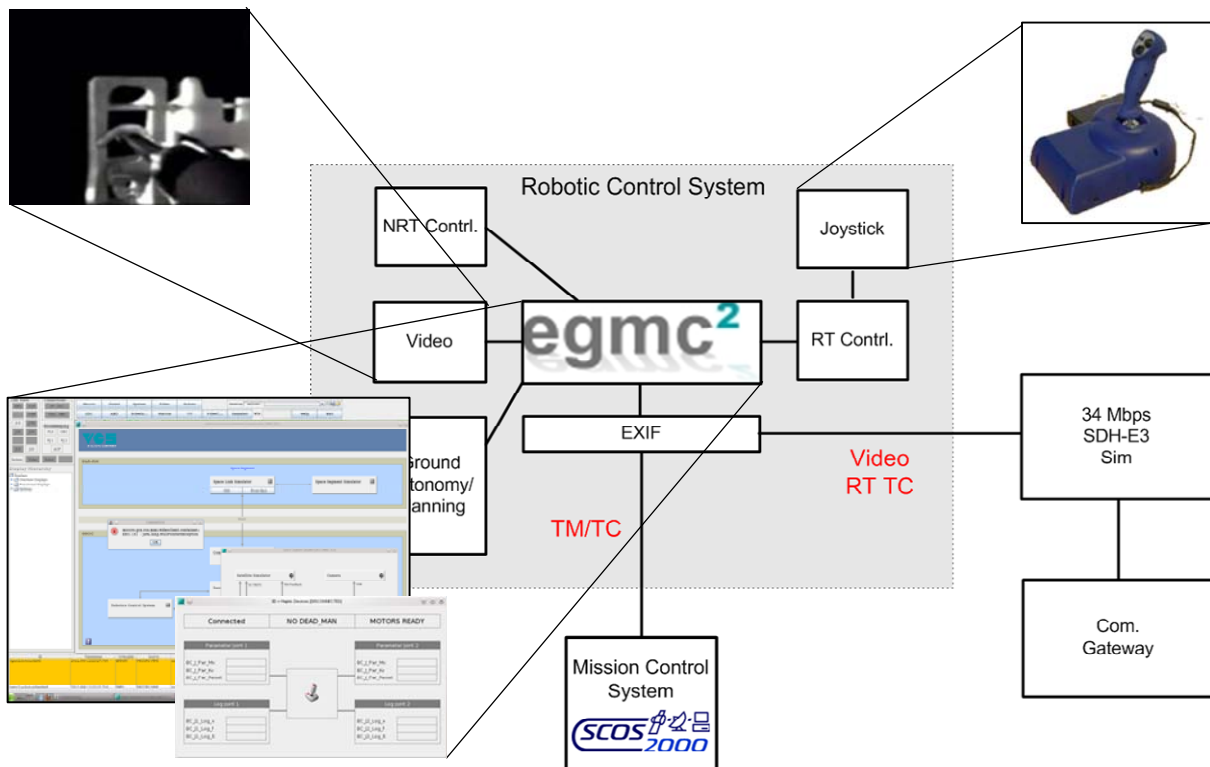


Abbildung 22: Komponenten des Robotic Control Systems (RCS)

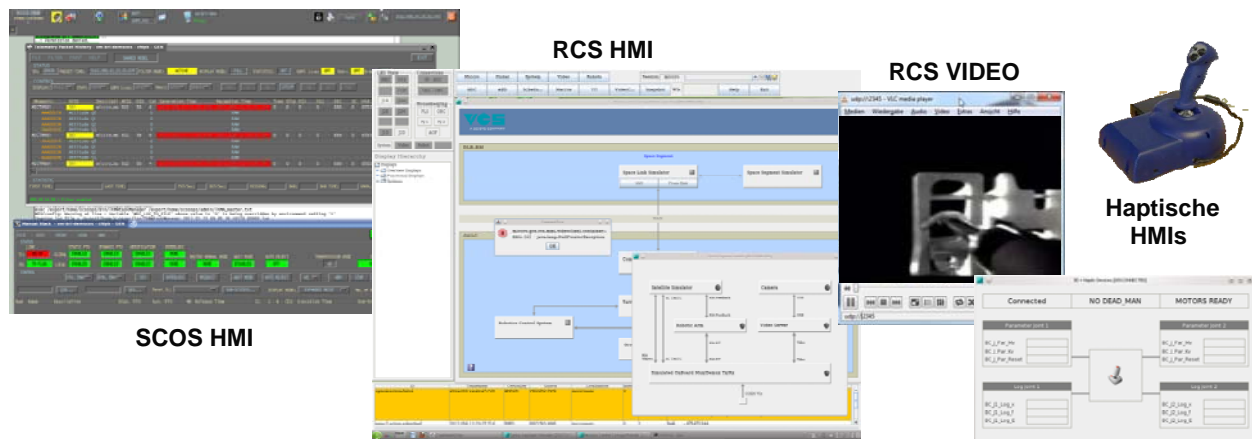


Abbildung 23: Komponenten der Benutzerschnittstelle

Abbildung 23 zeigt einen Ausschnitt der in der Benutzerschnittstelle des Robotic Control System zusammengefassten Komponenten. Diese sind:

- SCOS HMIs: Diese werden über einen X-Server auf einer eigenen Konsole gestartet
- egmc²-HMI: Dieses speziell auf die Bedürfnisse des RCS Operators zugeschnittene HMI wird lokal auf einer Workstation als Rich Client gestartet. Das HMI beinhaltet eine Vielzahl von weiteren HMIs, die dort zentral gestartet werden können:
 - Video Display zur Darstellung des Roboters und dessen Bewegungen
 - Generische Applikationen zur Anzeige von Telemetrie
 - Generische Applikationen zum Versand von Telekommandos
- SatViz: Visualisierung der Satellitenflugbahn und –lage
- SatMon: Darstellung der mittels SCOS prozessierten Telemetrie in konfigurierbaren Listen, grafischen Darstellungen und Blockschaltbildern

3.1.2.1.1 Ground Autonomy

Für die Implementierung einer bodenbasierten Autonomiekomponente wurde ein System basierend auf einer Problem- und Umgebungsbeschreibung in der Problem Domain Definition Language (PDDL) gewählt. Für Probleme, welche in dieser Beschreibung zur Verfügung stehen existiert eine Vielzahl von unterschiedlichen Planungsanwendungen, die auf ihre Anwendbarkeit überprüft wurden.

Im Rahmen dieses Arbeitspaketes fand auch ein Workshop unter Einbeziehung von SCISYS UK und des King's College London (Derek Long) statt. Hierbei wurde eine Vielzahl von Problemen bei der Beschreibung von Problemen in PDDL, bzw. in der Berücksichtigung verschiedener Beschreibungsmerkmale in der Planungssoftware, diskutiert. Hierdurch kann das entstandene Planungssystem nicht als generisch für eine Vielzahl von Problemen angesehen werden. Es ist jeweils eine Verwendung einer auf die Problemklasse abgestimmten Planungsapplikation erforderlich.

Für die Systemdemonstration wurden jedoch zwei Szenarien implementiert, die den Nutzen eines solchen Planungssystems verdeutlichen: einerseits zur Planung von auch parallel ablaufenden Aktivitäten und andererseits als Assistenzsystem zur Unterstützung des Operators.

In letzterem Beispiel wurde für eine telepräsenz OOS-Operation die Sequenz von erforderlichen Aktivitäten zeitlich modelliert. In dieses Planungssystem werden nun zusätzlich über die aus dem SCOS empfangene Telemetrie Statusinformationen (z.B. Batterieladezustände) eingefügt. Ergeben sich hieraus Hinweise, dass das Ziel der Telepräsenzphase (Einfangen eines Satelliten) innerhalb des fest vorgegebenen Zeitfensters nicht erreicht werden kann, wird eine Warnung an den Operator ausgegeben.

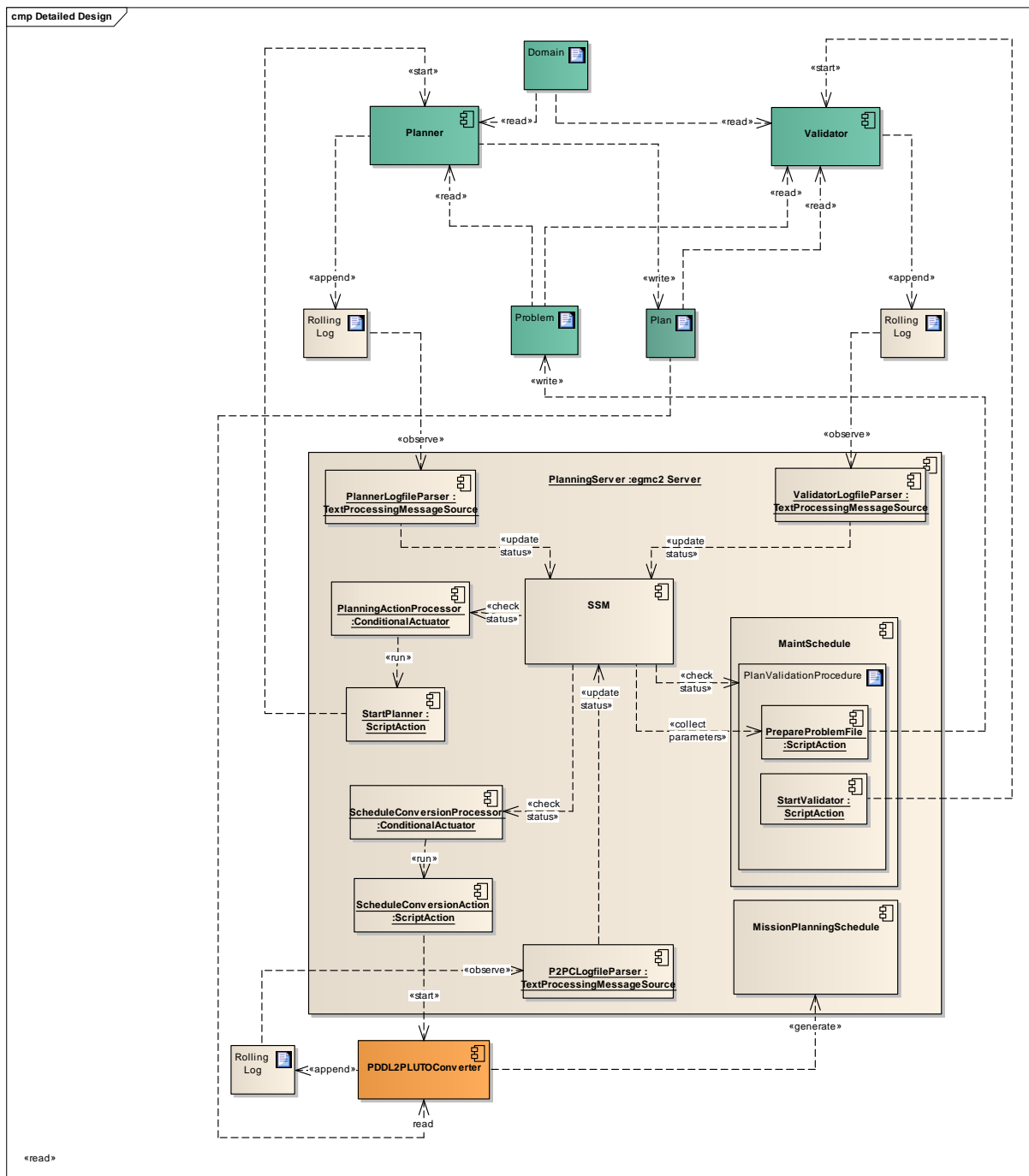


Abbildung 25: Orchestrierung des Planungsprozesses durch den RCS/Planning Server

Für die Steuerung der Prozesse können hier die durch das egmc²-Framework bereitgestellten Applikationen, etwa für das Absetzen von Kommandos der Generic Control Client (GCC), verwendet werden. Weiterhin ist zu sehen wie ein Planungsprozess manuell ausgelöst wird. Im Log-File ist zu erkennen, wie über externe Skripte der COTS-Planner verwendet wird und schließlich ein Plan als Lösung des Problems gefunden wird.

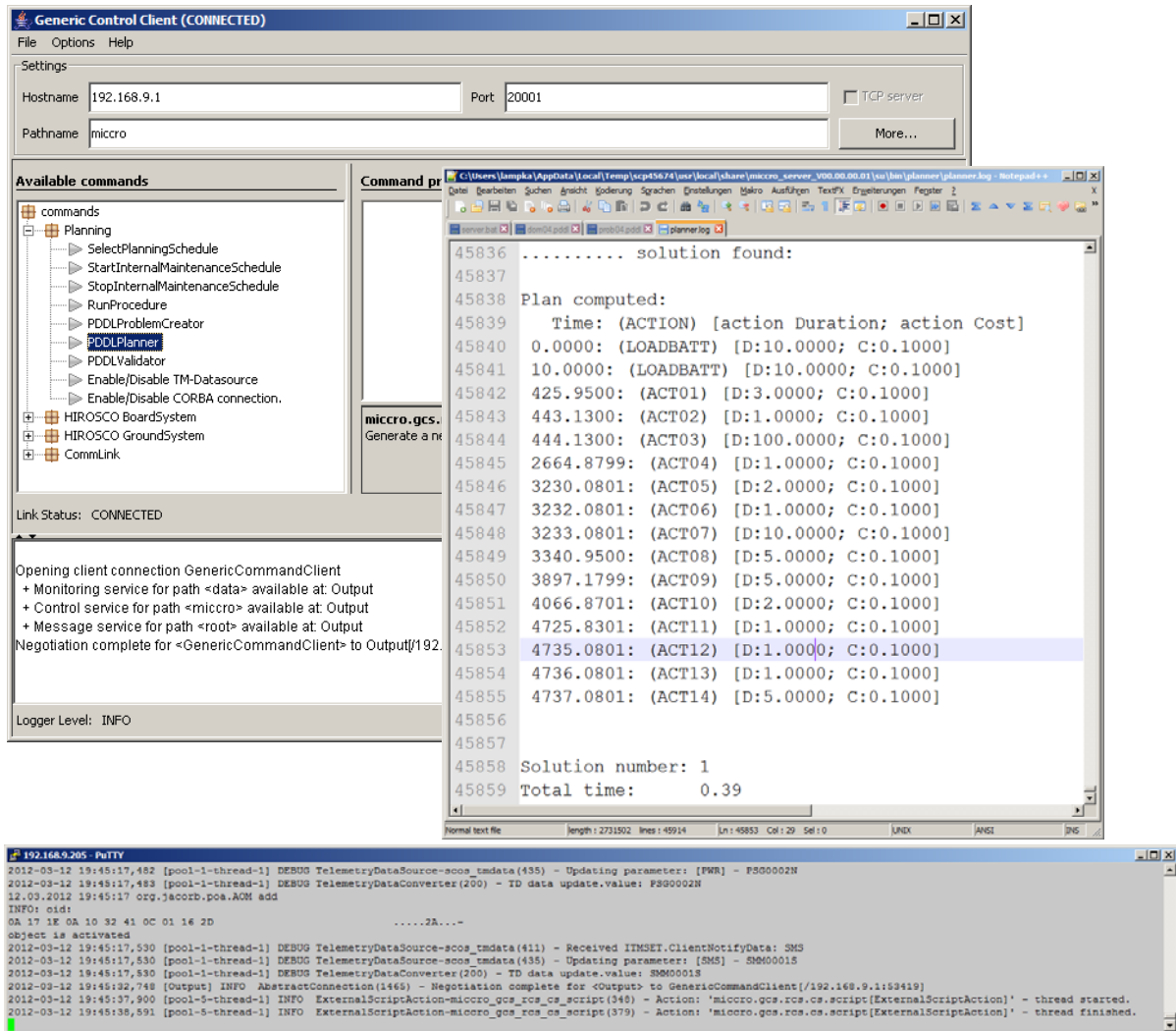


Abbildung 26: Mechanismen zur Steuerung des Planungsprozesses

3.1.2.1.2 MCS Interface

Die Schnittstelle zwischen dem egmc²-Framework-basierten Robotic Control System (RCS) und SCOS basierten Mission Control System (MCS) wurde auf der von SCOS bereitgestellten External Interface (EXIF) Schnittstelle implementiert. Diese Schnittstelle nutzt den Common Object Broker Request Architecture (CORBA) Standard. Für EXIF wurde in egmc² ein entsprechendes Modul entwickelt um den Datenaustausch zwischen RCS und MCS zu ermöglichen.

In der nachfolgenden Abbildung ist gezeigt wie die Telemetrie-Parameter aus SCOS (rechts) auch innerhalb der Generic Data Clients (GDC) im RCS verfügbar gemacht werden können. Auf diese Weise sind entweder die gesamte in SCOS verfügbare Telemetrie oder nur definierte Teile auch für die Überwachung im RCS darstellbar.

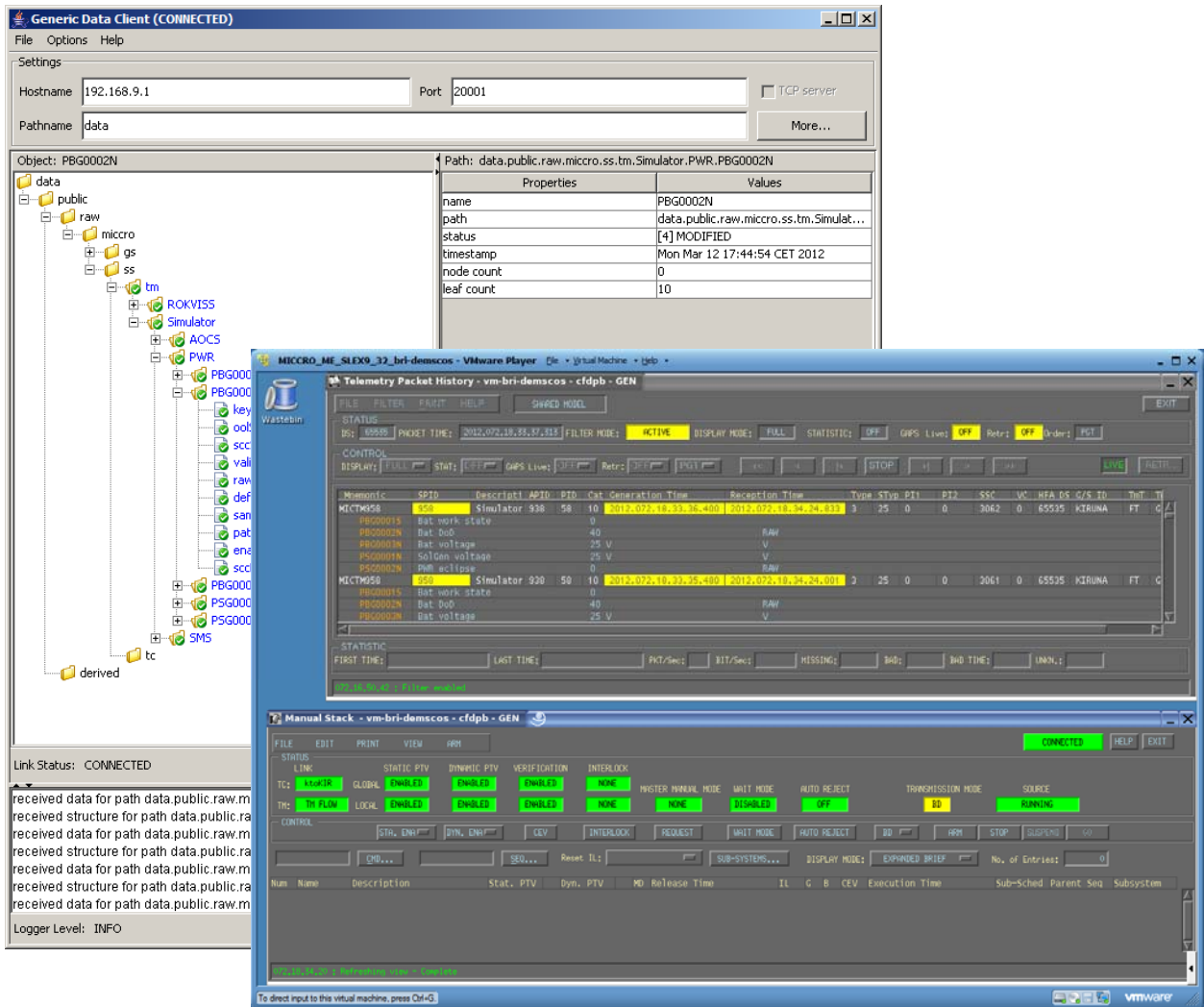


Abbildung 27: Empfang von SCOS basierter Telemetrie im egmc²-Framework

3.1.2.1.3 Simulatoren und Test Stubs

Um eine reibungslose Integration und auch einen frühen Test des Communication Gateways und des Onboard Multiplexers zu ermöglichen, wurden Simulatoren für die Schnittstelle zum Roboter und zum force-feedback Joystick implementiert (vgl. Abbildung 28). Dafür wurde der für VxWorks entwickelte Quellcode angepasst, so dass die Test Stubs in einer Linux-Umgebung lauffähig sind.

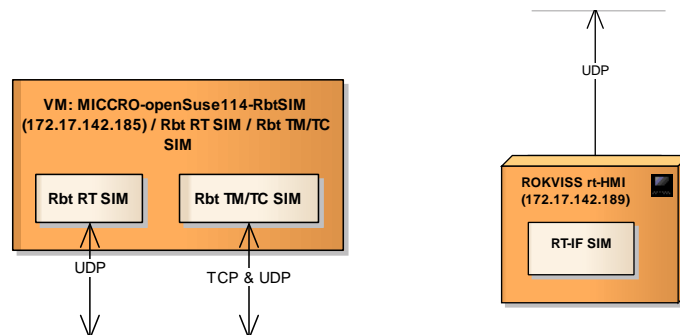


Abbildung 28: Simulationsplattform mit allen relevanten Schnittstellen während der Integration bei SCISYS in Bochum

Durch diese werden die folgenden Schnittstellen simuliert:

- Rbt RT ROKVISS: Realtime Datenpakete mit den Soll- und Ist-Daten für die Robotertelepräsenzsteuerung. Diese Schnittstelle wird mit einem auf einer Linux-Plattform lauffähigen Test Stub emuliert. Hierdurch wird zwar nicht die bis auf 2ms genaue Taktung der Datenpakete ermöglicht, aber eine der Realität entsprechende Datenrate.
- Rbt RT Joystick: Realtime Datenpakete mit den Soll- und Ist-Daten für die Robotertelepräsenzsteuerung. Diese Schnittstelle wird mit einem auf einer Linux-Plattform lauffähigen Test Stub emuliert. Hierdurch wird zwar nicht die bis auf 2ms genaue Taktung der Datenpakete ermöglicht, aber eine der Realität entsprechende Datenrate.
- Rbt TM/TC ROKVISS: NRT TM & TC Daten zu und von der ROKVISS-Ansteuerung in CCSDS / PUS TM & TC Source Pakets, die im Onboard Multiplexer in freie Idle TM Frames, bzw, im Communication Gateway in generierte TC Frames eingepackt werden.
- Rbt TM/TC Joystick: Die NRT TM der Regelungssoftware für die Telepräsenzschleife wird über die vom egmc²-Framework bereitgestellte MCC-API direkt bereitgestellt. Über diese werden auch die Regelparameter kommandiert.

Die entsprechenden Simulatoren wurden anschließend während der Integration Schritt für Schritt durch die eigentlichen Schnittstellen zur ROKVISS-Steuerungssoftware, der Regelungssoftware für den Joystick mit Krafrückkopplung und der eigentlichen Hardware ersetzt. Die nachfolgende Abbildung, welche aus Abbildung 18 extrahiert wurde, veranschaulicht diese Schnittstellen noch einmal.

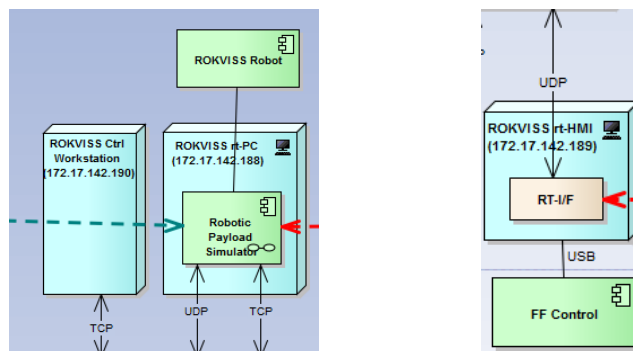


Abbildung 29: Getrennte Plattformen für die Ansteuerung des ROKVISS-Roboters (links) und des Joysticks (rechts) während der Integration bei RMC und GSOC in Oberpfaffenhofen

3.1.2.1.4 Anpassung der ROKVISS NRT M&C-Schnittstelle

Um die entsprechenden dem Design des Demonstrators angepassten Schnittstellen bereitzustellen, waren Anpassungen aller bisher verwendeten Schnittstellen zur ROKVISS Hardware notwendig. Es erfolgte eine Umstellung der bereits teilweise PUS konformen NRT Source Pakets auf eine für beide SCOS Versionen kompatible Implementierung. Hierbei war eine enge Zusammenarbeit der Projektpartner während der Systemintegrationsphasen nötig. Diese Umstellung begründet sich darin, dass der PUS unterschiedliche Konfigurationen seiner Pakete erlaubt, wobei SCOS eine andere Konfiguration verlangte als beim DLR-RMC verwendet wurde.

Durch diese Änderung wird eine mehr dem CCSDS-Standard folgende Implementierung verwendet, mit welcher die TM/TC-Daten der Onboard-Komponenten über das SCOS-basierte MCS zusammen mit den TM/TC-Daten des Raumfahrzeugs prozessiert werden können. Abgesehen von den RT-Datenpfaden wird kein zusätzlicher proprietärer Datenkanal mehr benötigt.

Während der Implementierung waren einige Iterationen notwendig, bis eine Kompatibilität zwischen dem CCSDS / PUS Standard und den beiden verwendeten SCOS Versionen 5.1 und 3.1 (GSOC) gewährleistet werden konnte. Dies ist u.a. auf eine nicht immer vollständige Unterstützung des Standards, z.B. für Strings variabler Länge, zurückzuführen.

3.1.2.2 Mission Control Systems (MCS)

Das Mission Control System (MCS) ist die für die Überwachung und Steuerung der Mission verantwortliche Software- und Hardwarelösung. Hierüber werden Verbindungen zu Bodenstationen oder einem Netz aus Bodenstationen hergestellt.

Aus diesem Grund müssen Telemetriedaten als Datenstrom identifiziert in virtuellen Kanälen empfangen werden, wie sie in den entsprechenden Standards definiert sind. Dies gilt entsprechend auch für den Versand von Telekommandos an das ausführende NCTRS/NIS-System. Innerhalb des MICCRO Projektes wurde für die das MCS SCOS 2000 der ESA eingesetzt, wie es innerhalb des GSOC auch schon für andere Missionen im Einsatz ist und bei europäischen Raumfahrtmissionen als Standard betrachtet werden kann.

Neben der im GSOC verwendeten, auf SCOS Version 3.1 basierenden, wurde auch eine Realisierung auf Basis der SCOS Version 5.1 erfolgreich getestet.

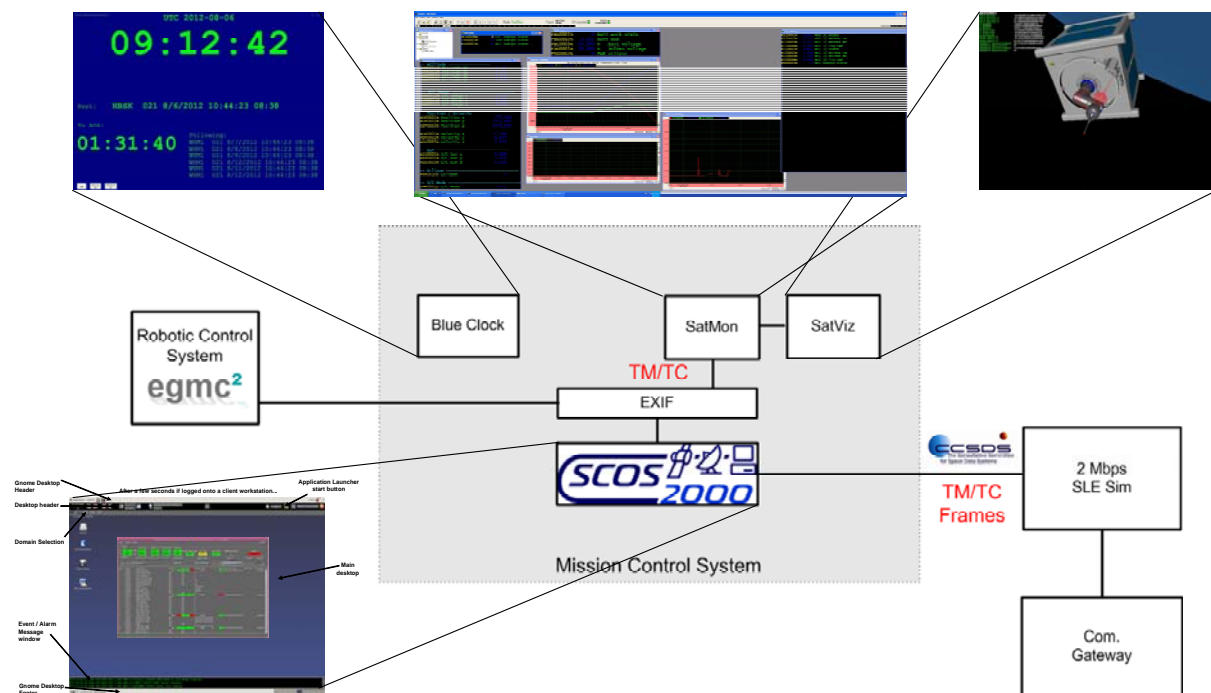


Abbildung 30: Komponenten des Mission Control Systems (MCS)

3.1.2.2.1 MCS-Prototyp (SCOS 5.1)

Wie innerhalb des Projektes vorgesehen, wurde durch SCISYS ein Mission Control System (MCS) als virtuelle Maschine aufgesetzt. Als Basis hierfür wurde das von der ESA bereitgestellte SCOS 5.1 (auch SCOS 2000 oder SCOS2K) verwendet.

Neben einigen grundlegenden Konfigurationen basiert die Installation der SCOS-Instanz auf der Erstellung einer so genannten Mission Information Base (MIB). Zur Erstellung dieser MIB wurde eine Applikation entwickelt, die in mehreren Schritten eine einheitliche Systemkonfiguration für das Projekt MICCRO generiert.

Grundlage hierfür ist eine Beschreibung der Telemetrieparameter sowie der Telekommandos für die über das HIROSCO Framework bereitgestellten ROKVISS-Daten in einem Excel-Arbeitsblatt. Hierin sind die einzelnen Parameter und die Struktur der TM Source Pakets beschrieben. Eine analoge Beschreibung existiert für die Parameter wie sie über den Simulator der Satellitenplattform und das darin eingebettete Orbitmodell bereitgestellt werden. Zusätzlich werden alle Kommandos wie sie an ROKVISS Onboard und Onground, den Simulator der Satellitenplattform und auch intern im egmc²-Server versendet werden sollen in einer vorgegebenen XML-Struktur (Command Dictionary) definiert. Der Konverter erzeugt hieraus automatisch die egmc²-Konfigurationsdateien für den RCS/Planning Server, die MIB welche für beide SCOS Versionen (5.1 und 3.1 (GSOC)) importiert werden kann.

Weiter wird diese MIB auch verwendet um den Simulator der Satellitenplattform (GSTVi/SIMSat) zu konfigurieren.

Durch dieses Vorgehen wird eine konsistente Konfiguration des Gesamtsystems gewährleistet. Zusätzlich werden nur wenige zusätzliche Anpassungen wie etwa die Erstellung von Command Stacks in SCOS oder Kommandomakros in egmc² benötigt, um eine vollständige Konfiguration zu erstellen.

Um einen Einstieg in die Installation, Konfiguration und Bedienung von SCOS 5.1 und GSTVi zu erhalten, hat schon Ende 2011 ein 2-tägiger Workshop mit einem Kollegen von SCISYS UK stattgefunden.

3.1.2.2.2 MCS-Prototyp (SCOS 3.1)

Analog zu dem von SCISYS implementierten Mission Control System wurde im Rahmen dieses Arbeitspaketes eine weitere Instanz am GSOC implementiert basierend auf der innerhalb des GSOC verwendeten SCOS Version 3.1 (GSOC), welche spezielle Korrekturen, Erweiterungen und Anpassungen enthält.

Für die Konfiguration wurde die gleiche von SCISYS bereitgestellte MIB verwendet. Hierbei traten immer wieder kleinere Inkompatibilitäten auf, die aber in mehreren Iterationen weitestgehend ausgeräumt werden konnten.

Für die Command Stack Files musste ein Konverterskript erstellt werden, um die SCOS 5.1 Versionen entsprechend in das für SCOS 3.1 gültige Format zu überführen.

3.1.2.2.3 SatMon

Für die im GSOC eingesetzte SCOS Version 3.1 (GSOC) existiert mit SatMon ein eigenständiges TM-Visualisierungssystem, welches als SCOS-Frontend für die TM-Darstellung genutzt wird. Dieses wurde schon erfolgreich mit anderen M&C-Systemen, bspw. RAMSES, zur besseren Überwachung und Darstellung der Telemetrie genutzt. SatMon wurde für seine Verwendung im Rahmen der MICCRO Demonstration vorbereitet und konfiguriert. Diese Software nutzt ebenfalls die EXIF-Schnittstelle, um die Telemetrie aus dem Datenbestand des SCOS abzufragen.

3.1.2.2.4 SatViz

Die Visualisierung der Satellitenbahn und -lage erfolgt aufbauend auf den in SatMon vorliegenden Parametern. Diese werden im nachfolgend beschriebenen Orbitmodell erzeugt, in den durch GSTVi bereitgestellten TM Source Pakets einbezogen und über den Onboard MUX und das Communication Gateway in das SCOS-basierte MCS geroutet.

SatViz wird genutzt um die Bewegung des Roboterarms anhand der NRT Telemetrie sowie die simulierte Flugbahn darzustellen. Das nachfolgende Bild verdeutlicht wie der ROKVISS Roboter auf einer Satellitenplattform installiert ist. Dieser Satellit wird in der Demonstration als fiktiver OOS Servicer kontrolliert.



Abbildung 31: Fiktiver MICCRO OOS Servicer

3.1.2.3 Communication Gateway (CommGW) und Onboard Multiplexer (Onboard MUX)

Als zentrale Komponente für die Verwaltung der einzelnen Datenströme im Demonstrationssystem wurde die so genannte Gateway Application implementiert. Grundlegend für die Implementierung dieser Komponente war hierbei das Schnittstellendesign des Communication Gateways und des Onboard Multiplexers. Beide Komponenten basieren auf der Gateway Application und werden nur durch Konfigurationsänderungen auf die jeweiligen Anforderungen angepasst.

Aufgrund der sehr hohen Anforderungen an ein echtzeitnahes Verhalten der Applikation wurde eine sehr performante und optimierte Implementierung in C++ durchgeführt. Als Plattform kommt jedoch eine herkömmliche, nicht auf Echtzeitverhalten optimierte Linux-Installation zum Einsatz. Dies ist durch die Einschätzung bedingt, dass auch auf einer solchen Plattform bei den zu erwarteten Datenströmen von insgesamt kleiner 10MBit/s ein sehr gutes Zeitverhalten zu erwarten ist. Die nachfolgende Abbildung verdeutlicht die Vielzahl der Schnittstellen zu allen Systemkomponenten:

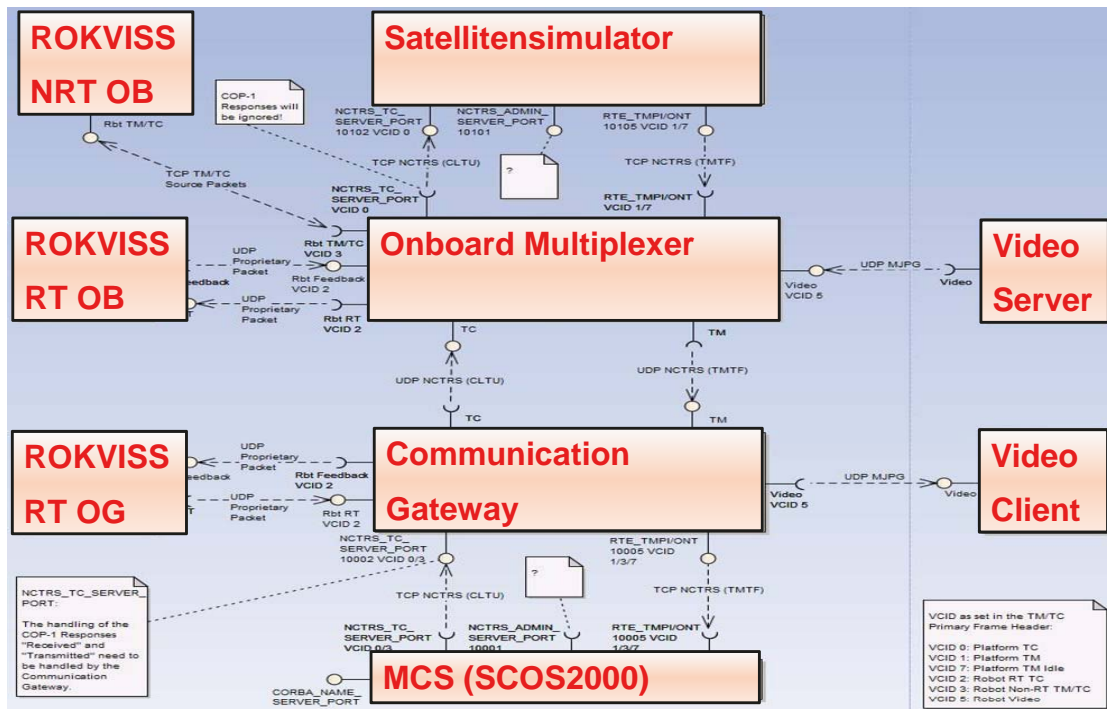


Abbildung 32: Schnittstellen des Communication Gateways (CommGW) und des Onboard Multiplexers (Onboard MUX)

Der ebenfalls auf der Quellcodebasis des Communication Gateways basierende Onboard MUX kann durch Konfiguration als inverse Komponente Communication Gateway betrieben werden.

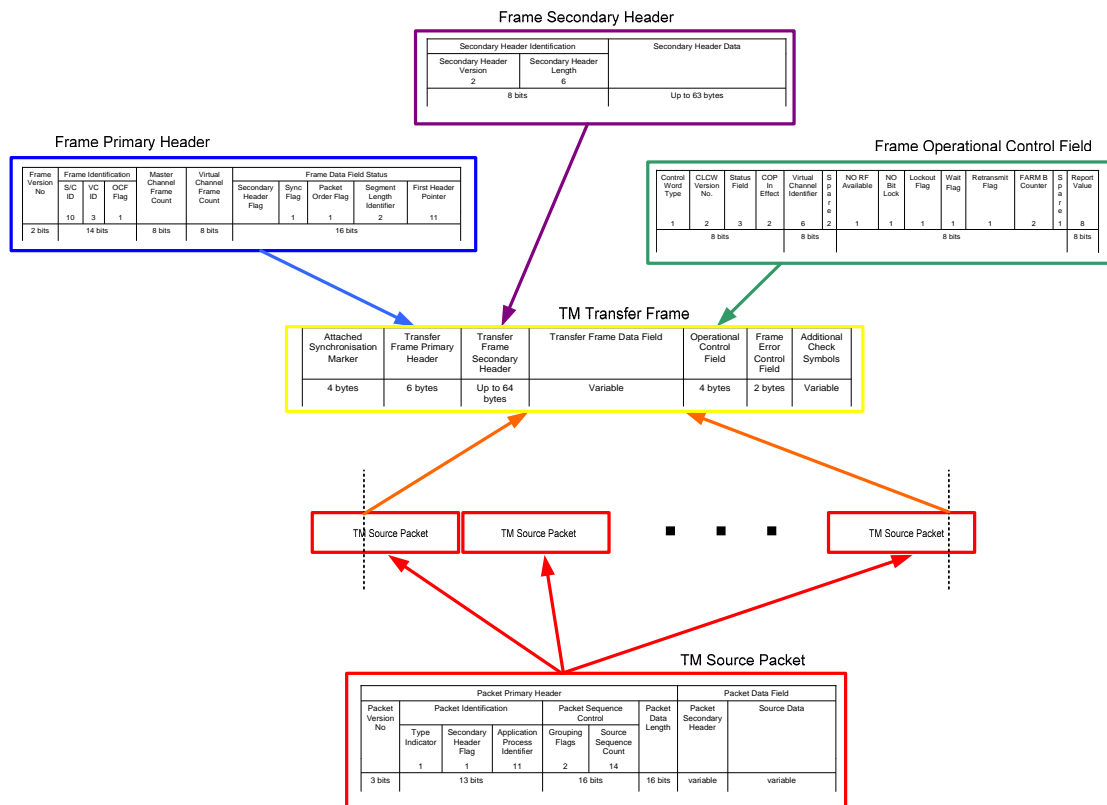


Abbildung 33: Protokollstruktur (Onboard MUX → CommGW)

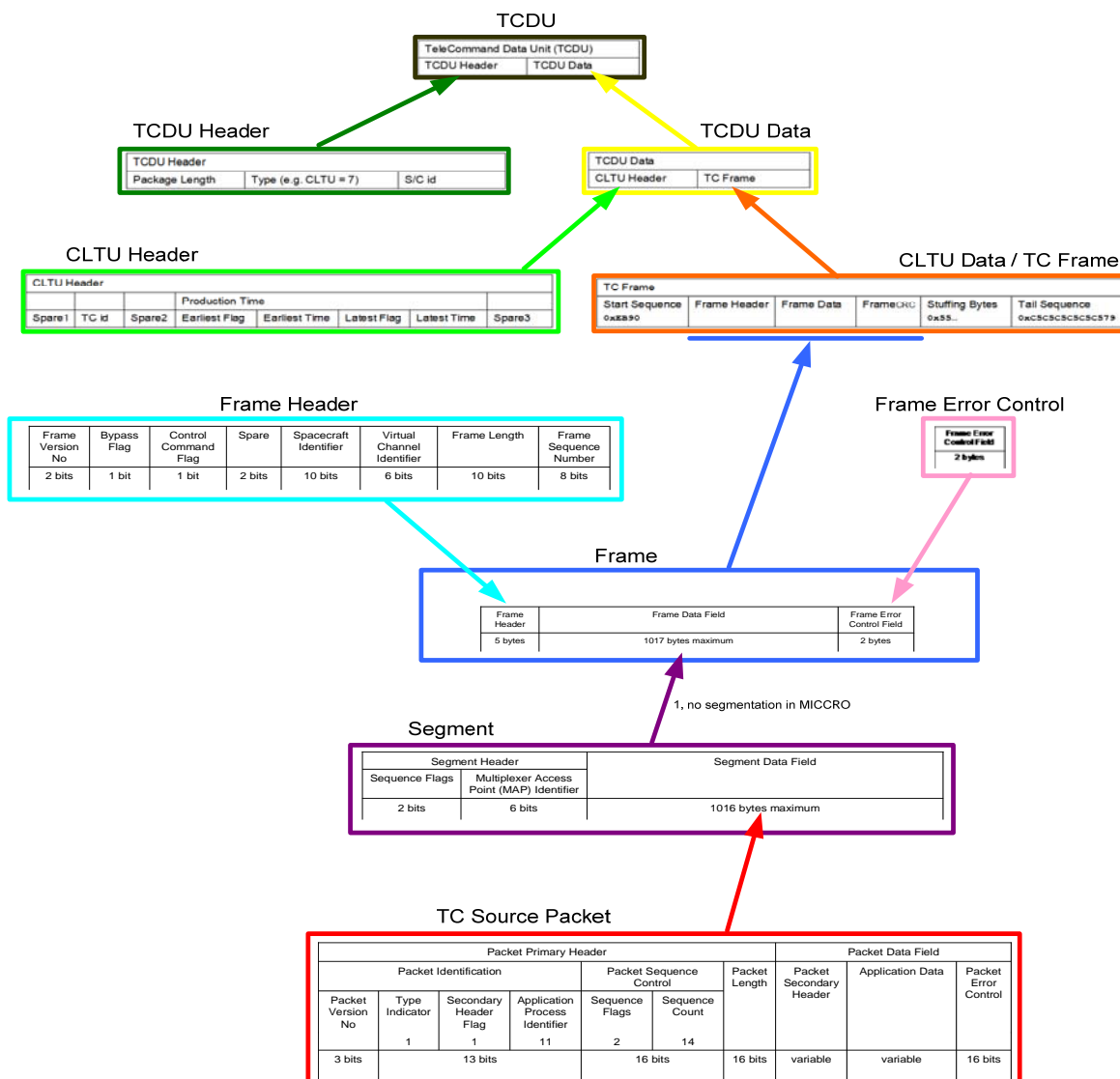


Abbildung 34: Protokollstruktur (CommGW → Onboard MUX)

3.1.2.4 Linksimulatoren

Für die unterschiedlichen terrestrischen und funkbasierten Übertragungstrecken (inkl. Link zum Raumfahrzeug) kommen Simulatoren basierend auf WANem zum Einsatz. Hierdurch können auf den unterschiedlichen Übertragungstrecken Latenz, Jitter und Paketverluste für IP-basierte Netze simuliert werden. Zwei unterschiedliche terrestrische Übertragungstrecken zwischen Bodenstation und Kontrollzentrum (SLE Netzwerk für TM/TC und ein SDH/E3 Link für Video und RT TM/TC) werden simuliert. Für den Weltraumlink können ebenfalls die Charakteristiken angepasst und deren Auswirkungen untersucht werden.

3.1.2.5 Robotic Payload Simulator

Schließlich zeigt Abbildung 35 Fehler! Verweisquelle konnte nicht gefunden werden. die Details zum Robotic Payload Simulator, also dem Teil der Software, der die robotische Nutzlast an Bord des Satelliten simuliert. Die Software basiert dabei auf dem HIROSCO-Framework und wird zum Beispiel in ähnlicher Form im Labor vom DLR-RMC zur Demonstration des ROKVISS Experiments eingesetzt. Der HIROSCO Supervisor übernimmt darin die Überwachung und Steuerung aller angebotenen Komponenten. Die TM/TC Komponente ist für den Austausch der TM/TC Daten mit dem Onboard MUX verantwortlich. Die Robotersteuerung implementiert die Sercos II Schnittstelle zum ROKVISS

Engineering Modell und dessen Impedanzregelung. Der Roboter kann entweder über die Task Oriented Programming Komponente tele-operiert oder über die bilaterale Regelung im Tele-präsenz Modus betrieben werden. Letztere tauscht dafür Daten mit dem Onboard MUX im Echtzeittakt von 2 Millisekunden aus. Der Simulator wird dabei auf eine Workstation mit VxWorks 6.9 betrieben.

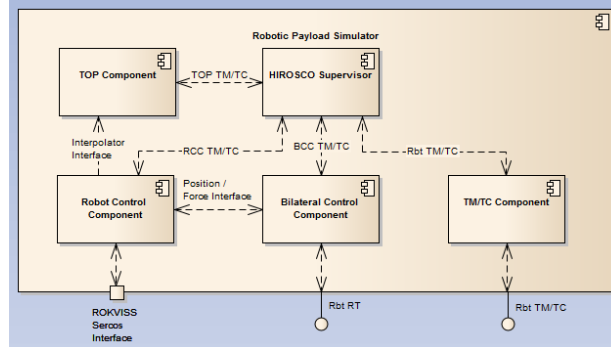


Abbildung 35: Design des Robotik Payload Simulators

3.1.2.6 Satellite Platform Simulator

Für die Simulation der Satellitenplattform wurde das zusätzliche Tool GSTVi der ESA eingesetzt, welches es erlaubt die Schnittstelle zwischen SCOS 200 und weiteren Systemkomponenten zu simulieren. Diese leistungsfähige Software wurde in einer Minimalkonfiguration installiert, um auf gesendete commands zu reagieren und ein en einfachen Satz Telemetriedaten zu erzeugen. Diese Telemetrie wird durch ein Satellitenmodell erzeugt, das einen LEO-Orbit beschreibt wie er für einen OOS-M9ission relevant wäre. In der Simulation sind Satellitenplattform und der ROKVISS-Roboter als Robotic Payload Simulator nicht gekoppelt, da für den Demonstrator eine feste Basis angenommen wird.

3.1.2.6.1 GSTVi

Für den MICCRO-Demonstrator wurde ein Satellitensimulator basierend auf dem GSTVi-Framework der ESA implementiert. Die Installation erfolgte in einer virtuellen Plattform, so dass ein Transport für die verschiedenen später beschriebenen Integrationsschritte problemlos möglich ist. Für die Konfiguration des Simulators wurden Erfahrungen wie sie bei SCISYS UK vorliegen genutzt. Die für SCOS erstellte MIB kann nach mehreren Iterationen und Verbesserungen nun auch für einen direkten Import in GSTVi genutzt werden.

Zusätzlich zu den rudimentären Fähigkeiten von SIMSat in Bezug auf die Bereitstellung von Telemetrie wurde ein eigenes, funktional sehr vereinfachtes Satellitenmodell integriert. Diese erlaubt die Bereitstellung von grundlegenden ACS-Parametern wie die der Satellitenbahn und -lage. Somit werden nun auch die für eine Visualisierung in SatViz notwendigen Telemetrie bereitgestellt.

3.1.2.6.2 Orbitmodell

Das weiter oben erwähnte Orbitmodell wurde als C++-Klasse seitens des DLR-GSOC entwickelt und bereitgestellt.

Die Integration in GSTVi sowie die Definition der Schnittstelle und Bereitstellung aller Eingabeparameter wurde durch SCISYS implementiert. Das Orbitmodell ist in der Lage die folgenden Parameter bereitzustellen:

```
time_t m_t_system; ///< system time in seconds since January 1, 1970 00:00
time_t m_t_obt;    ///< onboard time
ScMode m_mode_sc; ///< spacecraft mode
double m_q0;      ///< attitude quaternion element 1
double m_q1;      ///< attitude quaternion element 2
double m_q2;      ///< attitude quaternion element 3
double m_q4;      ///< attitude quaternion element 4
double m_om0;     ///< angular velocity vector component 1
double m_om1;     ///< angular velocity vector component 2
double m_om2;     ///< angular velocity vector component 3
```

```
bool m_eclipse_sc; //< spacecraft in eclipse?
double m_v_Sun_x; //< Cartesian Sun vector component 1
double m_v_Sun_y; //< Cartesian Sun vector component 2
double m_v_Sun_z; //< Cartesian Sun vector component 3
bool m_bat_chrg_status; //< battery charge indicator
double m_bat_dod; //< battery depth of discharge
double m_voltage_bat; //< battery voltage
double m_voltage_sp; //< solar panel voltage
bool m_eclipse_sp; //< solar panels eclipse flag
double m_v_pos_x; //< Cartesian position vector component 1
double m_v_pos_y; //< Cartesian position vector component 2
double m_v_pos_z; //< Cartesian position vector component 3
double m_v_vel_x; //< Cartesian velocity vector component 1
double m_v_vel_y; //< Cartesian velocity vector component 2
double m_v_vel_z; //< Cartesian velocity vector component 3
CCSDS::CUC m_t_system_cuc;
unsigned int m_t_system2000; //< system time in seconds since Jan 1, 2000 00:00
```

3.1.3 Systemverifikation

Für die Verifikation des Missionsbetriebskonzeptes wurden Szenarien definiert, die während der Systemdemonstration durchgeführt wurden, um die unterschiedlichen konzeptionellen Aspekte abzudecken:

- Darstellung der grundlegenden Funktionsblöcke
- Illustration der organisatorischen Abläufe
- Test der implementierten SW-Komponenten
- Überprüfung und Verifikation der aufgestellten Anforderungen

Die einzelnen Szenarien sind auf realistische Kontaktzeiten zugeschnitten und wurden wie folgt durchgeführt:

- Pass 1:
 - Szenario 1: Standardsatellitenbetrieb
- Pass 2:
 - Szenario 2: Übergabe in den Robotikbetrieb
 - Szenario 3: Initialisierung des Robotikbetriebs
- Pass 3:
 - Szenario 4: Telepräsenzbetrieb
- Pass 4:
 - Szenario 5: Supervised Autonomy
- Pass 5:
 - Szenario 6: Beendigung des Roboterbetriebs

In den folgenden Abschnitten werden genauere Informationen zu den Rahmenbedingungen der Demonstration, den eingebundenen Rollen, Aktivitäten während eines Satellitenpasses und auch den einzelnen Demonstrationsszenarien bereitgestellt.

3.1.3.1 Randbedingungen

Die folgenden Randbedingungen einer realen robotischen Satellitenmission im niedrigen Erdorbit (Low Earth Orbit, LEO) wurden im Vorfeld der Systemdemonstration zwischen den Verbundpartnern abgestimmt und um demonstrationsspezifische Elemente ergänzt:

- Ein Bodenstationskontakt, d.h. die Zeitdauer zwischen dem ersten Empfang des Satellitensignals (Aquisition Of Signal, AOS) und dem Abreißen des Empfangs (Loss Of Signal, LOS), dauert ca. 10 Minuten. Diese Zeitdauer wird für jeden Kontakt im Rahmen der Demonstration verwendet.
- Es wird eine mindestens fünfminütige Unterbrechung zwischen dem Ende eines Kontaktes (LOS-Ereignis) und dem Beginn des nächsten Kontakts (AOS-Ereignis) vorgesehen, um die Zeit zwischen den Kontakten in der Demonstration möglichst gering zu halten. In der Realität wäre hier mit längeren Zeiten ohne Kontakt zu Raumsegment zu rechnen.
- Ein Telepräsenzbetrieb des Roboters kann nur über einer dediziert dafür vorbereiteten Bodenstation erfolgen. In der Praxis wird das die Bodenstation Weilheim (WHM) sein.
- Eine zweite Bodenstation wird nur zur Vorbereitung der robotischen Aktivität und zur Videobeobachtung genutzt (z.B. Svalbard).
- Die Datenlinks sollen entsprechend der Sichtbarkeit und Anbindung der Bodenstationen (manuell) unterbrochen werden, falls der Demonstrator dies zulässt.
- Es wird die Rolle eines Simulationsmanagers (SIM-Manager) benötigt, der die Einstellungen vornimmt. Über das Kommandieren des GTSVi (TM on/off, TC on/off) kann hierbei die Verfügbarkeit der Funkverbindung simuliert werden.
- Es wird eine begleitende Präsentation auf den Wall-Screens zusätzlich zur einleitenden Präsentation gezeigt, die die jeweilige Anwendung illustriert.
- Es erfolgt im Telepräsenzbetrieb eine Warnung 2 und 1 Minute vor LOS durch den Flugdirektor.
- Anstatt alle Kommando-Stacks für die Aktivierung und Initialisierung des Roboters durch den RCS-Operator durchführen zu lassen, werden die ersten hierzu notwendigen Stacks bereits aus SCOS vom CMD-Operator abgeschickt. Dies ist nun auch in den weiter unten beschriebenen Abläufen während der Passes berücksichtigt.

3.1.3.2 Rollen

Flugbetriebsmannschaft (Flight Operations Team, FOT):

- 1 Flugdirektor (Mission Operations Manager, MOS)
- 1 Kommandooperator (Command Operator, CMD)
- 1 AOCS Subsystem Engineer
- 1 Netzwerkoperator (Network Controller, NC)
- 2-3 Nutzlastbedienung (Robotic Control System, RCS)

Simulationsteam:

- 1 Simulationsmanager (Vorbereiten und Überwachung der Demonstrationsumgebung; Zu- und Abschalten der Datenverbindungen)

3.1.3.3 Schematischer Ablauf eines Satellitenpasses

Dieses Unterkapitel beschreibt den grundsätzlichen Ablauf eines Kontakts eines im niedrigen Erdorbit fliegenden Satelliten mit einer Bodenstation und dem angeschlossenen Missionskontrollzentrums. Um dem Charakter der in der Studie als Referenz ausgewählten robotischen On-Orbit Service-Mission gerecht zu werden, wurde ebenfalls schon die Phase robotischer Aktivität (robotic activity phase) berücksichtigt. Eine weitergehende Detaillierung der für die Demonstration vorgesehenen Szenarien ist in einem nachfolgenden Unterkapitel (3.1.3.4) beschrieben.

- Der Ablauf eines Kontakts mit robotischen Nutzlastoperationen ist grundsätzlich wie folgt:
 - AOS -> Voice -> Freigabe zur Kommandierung -> etc. -> LOS (Details bei den Erklärungen der einzelnen Passes)

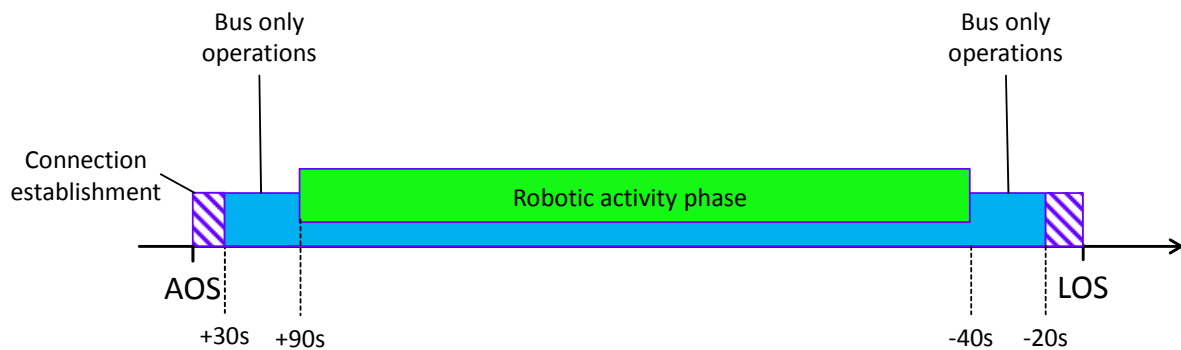


Abbildung 36: Schematischer Ablauf eines Satellitenkontakts

Der Beginn und das Ende werden durch die Ereignisse AOS und LOS markiert, welche für jeden Kontakt durch die Orbithöhe, die Position der Bodenstation auf der Erdoberfläche sowie die Überfluggeometrie (geringe maximale Elevation, d.h. „knapp“ über dem lokalen Horizont, oder hohe maximale Elevation, d.h. lokaler Zenitdurchgang) bestimmt sind.

3.1.3.4 Passes und Szenarien

Für die Kommandierung der robotischen Komponenten werden sogenannte CMD-Stacks verwendet, um komplexere Operationen, die aus mehreren Kommandos bestehen zu vereinfachen. Diese werden wie folgt in SCOS und auch im egmc² basierten RCS implementiert:

- Stack 0: Führt alle Subsysteme in den Zustand „INIT“
- Stack 1: Selektiert und aktiviert für jedes Subsystem die Konfiguration, Zustand „PREOP“
- Stack 2: Verbindet die Subsysteme untereinander
- Stack 3: Konfiguration wird geprüft
- Stack 4: Schaltet die Subsysteme in „SAFEOP“
- Stack 5: Schaltet alle Subsysteme operationell (Zustand „OP“, 8)
- Stack 6: Schaltet um von Teleoperation auf Telepräsenz
- Stack 7: Schaltet um von Telepräsenz auf Teleoperation
- Stack 8: Führt den Roboter in die Parkposition
- Stack 9: Führt den Roboter in die Startposition
- Stack 10: Roboter zieht nacheinander beide Federn
- Stack 11: Führt einen Soll-Ist-Abgleich für die Position durch
- Stack 12: Quittiert einen öfter auftretenden Fehler (z.b. bei Überlast) am Roboter
- Stack 13: Stoppt die zuletzt ausgeführte Aktion

3.1.3.4.1 Pass 1:

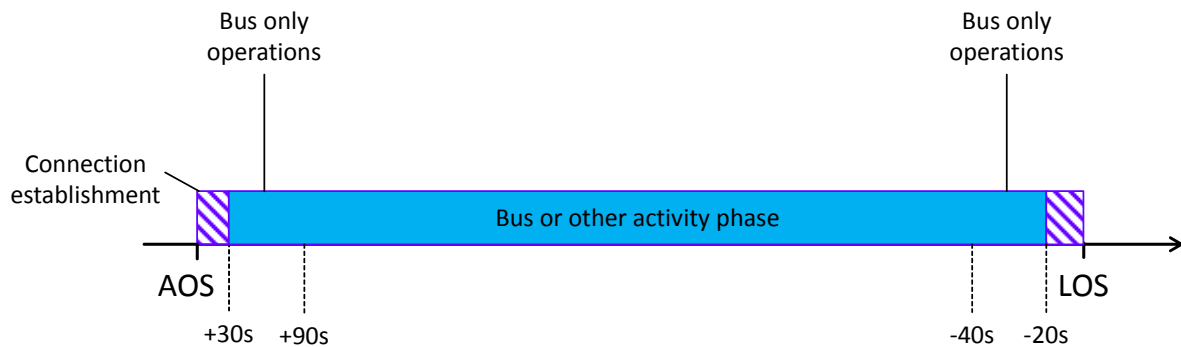


Abbildung 37: Pass 1: Bus Only Operation

3.1.3.4.1.1 Scenario 1: Standard Satellite Operations

- Dieser Pass soll genau der Definition wie in Kapitel 3.1.3.3, jedoch ohne robotische Aktivitäten, folgen.
- TM & TC mit den zwischengeschalteten MICCRO Komponenten
- Keine RT-Datenverbindung und robotische Aktivität
- 1 Pass (10 Minuten) als Einleitung und Erklärung des Setup
- MOS driven:
 - NC/MOS: Pre Pass Briefing
 - NC: Expected AOS -XX[min|secs]
 - NC/MOS: AOS Time
 - Subsystems/MOS: Quick System Check
 - NC/MOS: Green for CMD
 - Optional, um ggf. etwas Aktivität zeigen zu können:
 - MOS/CMD: S/C Mode Change and confirmation in TM. E.g. SCIENCE, which leads to unloading of the battery and later switch to SAFE and reloading. With speed factor 20 this will finish in one pass. Undo changes.
 - MOS/AOCS/CMD: ACS Mode Change Actuators on/off. Undo changes.
 - NC: Uplink down -XX[min|secs]

3.1.3.4.2 Pass 2:

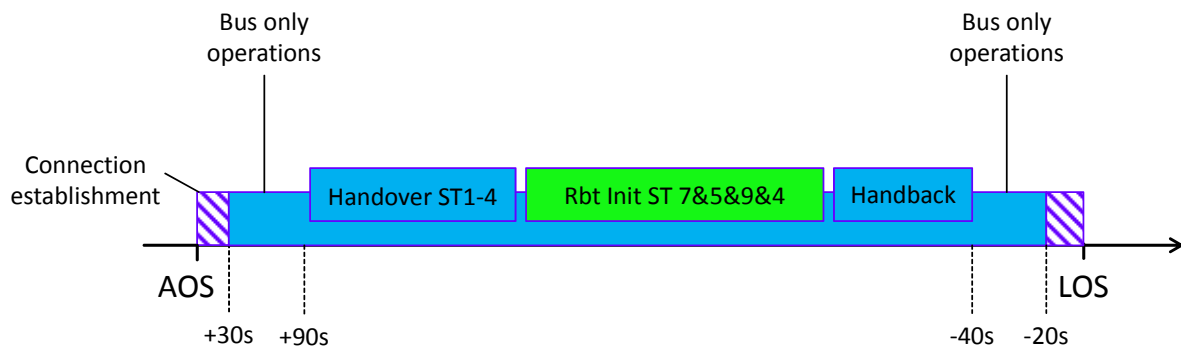


Abbildung 38: Pass 2: Handover to RCS and Initialisation

3.1.3.4.2.1 Scenario 2: Handover to robotic operation

- Übergabe an den RCS-Operator nach Abfrage aller Subsystem durch den Flugdirektor (MOS)
- Einfluss der Voice Loop wird deutlich
- MOS driven:
 - NC/MOS: Pre Pass Briefing
 - NC: Expected AOS -xx [min |secs]
 - NC/MOS: AOS Time
 - Subsystems/MOS: Quick System Check
 - NC/MOS: Green for CMD
 - MOS/AOCS/CMD: Execute STACK1-4
 - MOS/AOCS/CMD: deactivation of ACS actuators
 - MOS/RCS: Green for robotic operations

3.1.3.4.2.2 Scenario 3: Initialization of the robotic operation

- Initialisierung des Roboterarms
- RCS driven:
 - Execute STACK7 Teleoperation
 - RCS/MOS: Confirm: Execute STACK5 Operational
 - Execute STACK9 START POSITION
 - Execute STACK4 SAFE OPERATIONAL
 - MOS/RCS: Handback from robotic operations
- MOS driven:
 - MOS/AOCS/CMD: activation of ACS actuators
 - NC: Uplink down -xx[min |secs]

3.1.3.4.3 Pass 3:

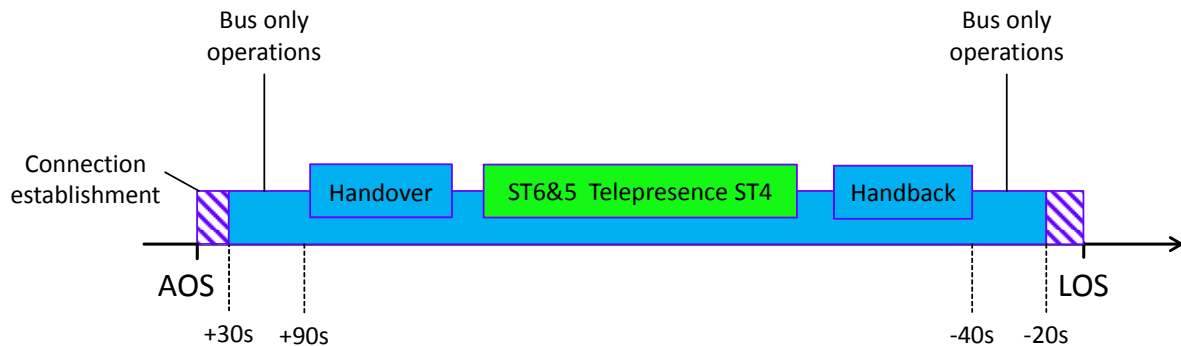


Abbildung 39: Pass 3: Telepresence Operation

3.1.3.4.3.1 Scenario 4: Telepresence operation

- Übergabe an den RCS-Operator nach Abfrage aller Subsystem durch MOS
- MOS driven:
 - NC/MOS: Pre Pass Briefing
 - NC: Expected AOS -XX[min|secs]
 - NC/MOS: AOS Time
 - Subsystems/MOS: Quick System Check
 - NC/MOS: Green for CMD
 - MOS/AOCS/CMD: deactivation of ACS actuators
 - MOS/RCS: Green for robotic operations
- RCS driven:
 - Execute STACK6 Telepresence
 - RCS/MOS: Confirm: Execute STACK5 Operational
 - Operate the robotik arm
 - Einfluss der Links mit Link SIM zeigen (Durch Paketverluste wird zum einen die Videoübertragung gestört und zum anderen die Telepräsenzsteuerung „hakelig“ durch das Einfügen einer Latenz wird die Telepräsenzsteuerung „weicher“ und das Video entsprechend verzögert. Eventuell können dies einzelne Teilnehmer selbst erfahren.)
 - Execute STACK4 SAFE OPERATIONAL
 - MOS/RCS: Handback from robotic operations
- MOS driven:
 - MOS/AOCS/CMD: activation of ACS actuators
 - NC: Uplink down -XX[min|secs]

3.1.3.4.4 Pass 4:

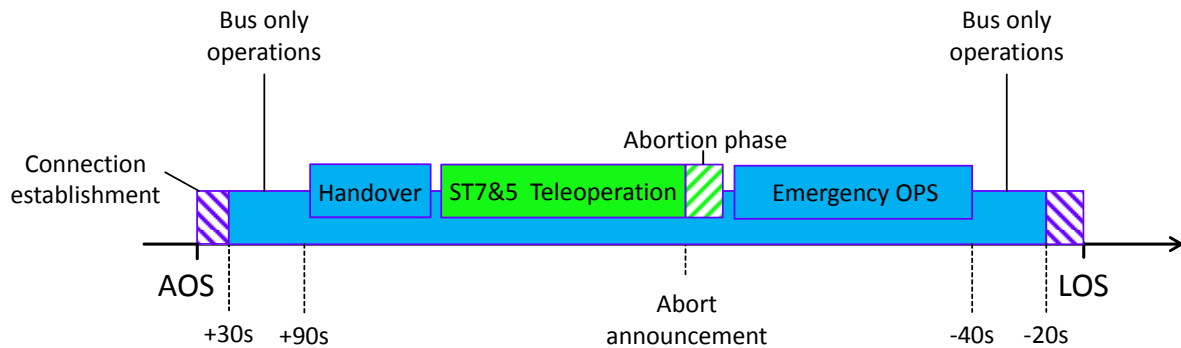


Abbildung 40: Pass 4: Supervised Autonomy and Abort Operation

3.1.3.4.4.1 Scenario 5: Supervised autonomy

- Übergabe an den RCS-Operator nach Abfrage aller Subsystem durch MOS
- MOS driven:
 - NC/MOS: Pre Pass Briefing
 - NC: Expected AOS -xxx[min |secs]
 - NC/MOS: AOS Time
 - Subsystems/MOS: Quick System Check
 - NC/MOS: Green for CMD
 - MOS/AOCS/CMD: deactivation of ACS actuators
 - MOS/RCS: Green for robotic operations
- RCS driven:
 - Execute STACK7 Teleoperation
 - RCS/MOS: Confirm: Execute STACK5 Operational
 - Execute STACK10 BOTH SPRINGS#
 - During execution press RED button triggering STACK8 HOME POSITION
 - RCS/MOS: Handback from robotic operations due to emergency situation
- MOS driven:
 - MOS/RCS: Decision on SAFE mode or resuming operations NORMAL mode
 - Execute STACK4
 - MOS/AOCS/CMD: activation of ACS actuators
 - NC: Uplink down -XX[min |secs]

3.1.3.4.5 Pass 5: Finalisation

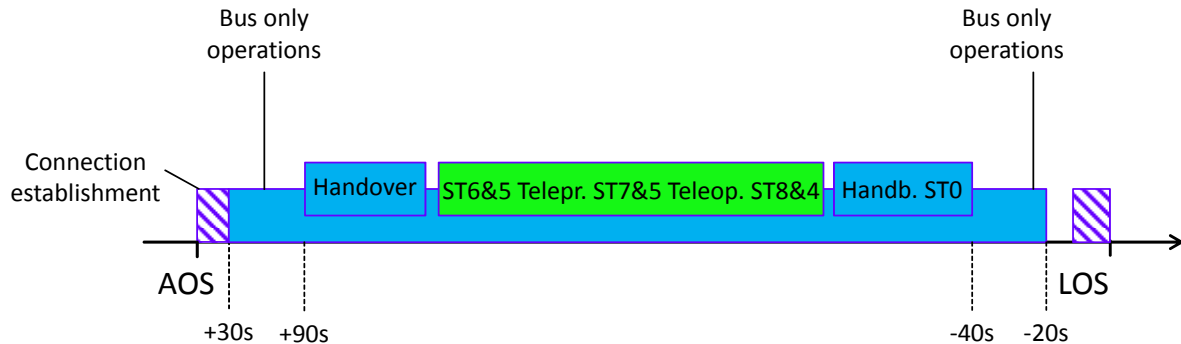


Abbildung 41: Pass 5: Finalisation of Robotic Operations

3.1.3.4.5.1 Scenario 6: Finalization of the robotic operation

- Übergabe an den RCS-Operator nach Abfrage aller Subsystem durch MOS
- MOS driven:
 - NC/MOS: Pre Pass Briefing
 - NC: Expected AOS -XX[min |secs]
 - NC/MOS: AOS Time
 - Subsystems/MOS: Quick System Check
 - NC/MOS: Green for CMD
 - MOS/AOCS/CMD: deactivation of ACS actuators
 - MOS/RCS: Green for robotic operations
- RCS driven:
 - Execute STACK6 Telepresence
 - RCS/MOS: Confirm: Execute STACK5 Operational
 - Operate the arm
 - Execute STACK7 Teleoperation
 - RCS/MOS: Confirm: Execute STACK5 Operational
 - Execute STACK8 HOME POSITION
 - Execute STACK4 SAFE OPERATIONAL
 - RCS/MOS: Handback from robotic operations
- MOS driven:
 - Execute STACK0
 - MOS/AOCS/CMD: activation of ACS actuators
 - NC: Uplink down -XX[min |sec]s

3.1.3.4.6 Offline: Ground Autonomy

Demonstration und Test der Ground Autonomy Komponente. Dies ist eine Offline Demonstration, die teilweise Unterstützung durch den CMD benötigt. Die unten genannten Werte für BatDoD können noch aktuell angepasst werden (müssen).

Vorbereitung:

- 1) Der egmc2-Server muss laufen (./runServer.sh)
- 2) Das Monitoring der planning und validation logs starten (./startMon.sh)
- 3) Der GenericControlClient (GCC) muss gestartet sein.

Durchführung:

- 1) Setzen des BatDoD auf einen Wert von 60% (über SCOS, **Test:changeParam**, BatDoD)
- 2) Invoke **PDDLProblemCreators** (über GCC)
- 3) Visuelle Kontrolle des Problems (über xterm -e 'head -n20 prob.pddl')
- 4) Invoke **PDDLPlanner** (über GCC)
- 5) Visuelle Kontrolle des Plans (über xterm -e 'less planner.log')
- 6) Invoke **PDDLValidators** (über GCC)
- 7) Visuelle Kontrolle des Reports (über xterm -e 'less validator.log')
- 8) Wiederholen des Schrittes 1) bis 7) mit geändertem BatDoD-Wert, bis Plan valide
Wenn Plan valide, dann:
- 9) Über das pdf-File das (valide) Ergebnis der Verifikation darlegen.
- 10) Invoke **StartInternalMaintenance** (über GCC)
- 11) Visuelle Kontrolle des Reports (über xterm -e 'less validator.log')
- 12) Setzen des BatDoD (über SCOS, **Test:changeParam**, BatDoD), bis Plan invalide
- 13) Über das pdf-File das (invalide) Ergebnis der Verifikation darlegen.
- 14) Invoke **PDDLPlanner** (über GCC)
- 15) Visuelle Kontrolle des Plans (über xterm -e 'less planner.log')
- 16) Warten, bis die Maintenance gelaufen ist.
- 17) Setzen des BatDoD (über SCOS, **Test:changeParam**, BatDoD), bis Plan valide

Ende:

- 1) Invoke **StopInternalMaintenance** (über GCC)
- 2) Schliessen der logging xterm Windows

Im Ergebnis soll gezeigt werden, dass das Planning auf einen Input-Wert dynamisch reagiert. Wenn ein Plan valide ist und sich im Verlauf der Zeit ein wichtiger Eingangsparameter ändert, wird der Plan in Bezug auf die Domain und das Problem invalide. Dann muss händisch eingegriffen werden und ein neuer Plan erstellt werden. Dieser Plan wird dann wieder als Grundlage des Domain/Problem/Plan-Triples genommen und immer wieder validiert (bei geändertem Eingangsparameter BatDoD). Die Änderung des Plans zeigt sich im Einfügen von SETPOWERSAVE-Phasen (in denen sich die Batterie erholen kann).

3.1.3.5 Impressionen von der Systemdemonstration

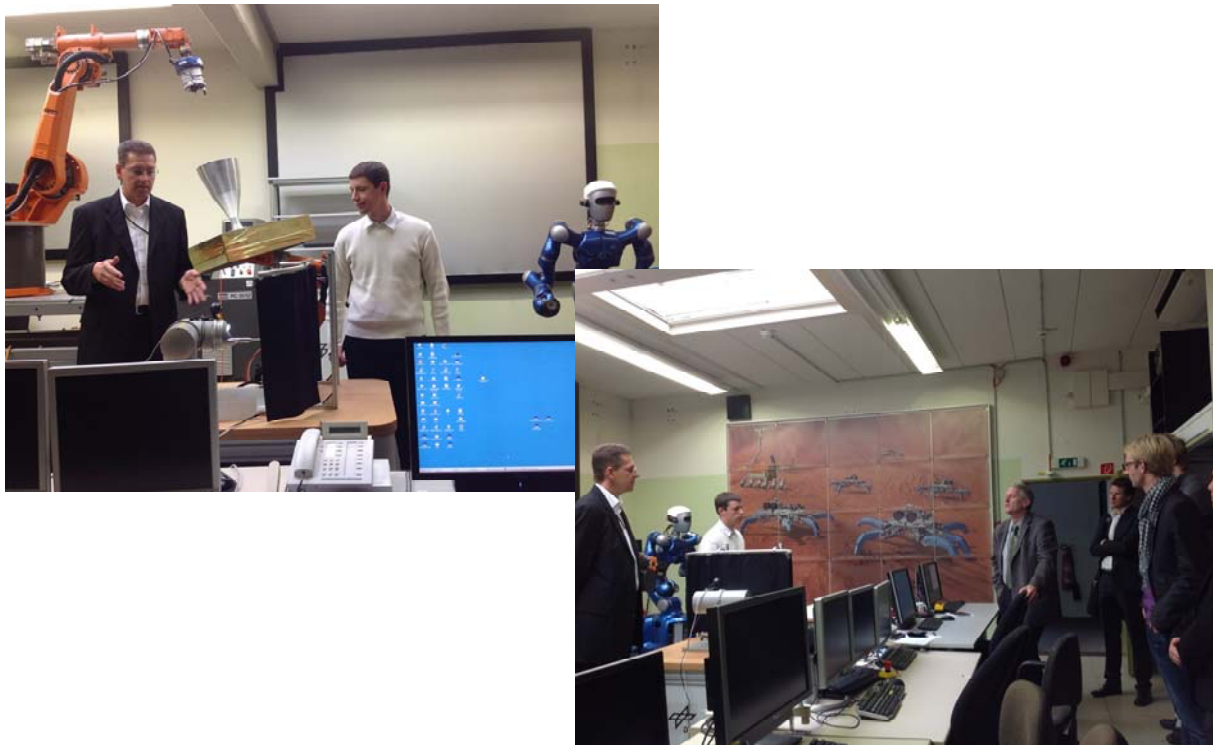


Abbildung 42: Simuliertes Raumsegment im Robotik-Labor des DLR-RMC



Abbildung 43: Durchführung des simulierten Missionsbetriebs im Kontrollraum des DLR-GSOC

3.2 Nutzen und Verwertbarkeit der Ergebnisse

Die Zielsetzung ein generisches Missionskontrollkonzept für robotische Missionen zu entwickeln und dieses exemplarisch anhand eines Prototypen zu demonstrieren wurde im Rahmen des Projektes erreicht und somit der folgende Nutzen für zukünftige Missionen und möglichen Kunden erzielt.

Konzepte für die Implementierung eines Missionsbetriebs, einer Kommunikationsinfrastruktur, einer Benutzerschnittstelle und einer bodengestützten Autonomie wurden erarbeitet und detailreich in den in Kapitel 2.3 vorgestellten technischen Dokumenten festgehalten. Diese konzeptionellen Ideen stehen für die Nutzung in zukünftigen Missionen zur Verfügung und können somit direkt in aktuellen Raumfahrtmissionen wie etwa DEOS eingebracht werden.

Die für den Systemdemonstrator prototypisch implementierten Softwarekomponenten können als Basis für eine weitere Implementierung für entsprechende Raumfahrtmissionen verwendet werden. Hier können insbesondere das Communication Gateway und die Robotic Control System herausgestellt werden.

Eine Verwendung der Ergebnisse in zukünftige Missionen und Analogmissionen ist wie folgt möglich:

- Ergänzende Analyse der Kommunikationsinfrastruktur
 - SLE-Netzwerk zu den Bodenstationen
 - Einbindung von Empfangs- und Sendehardware (z.B. CORTEX)
 - DTN ?
 - Einbeziehung realer Übertragungsstrecken
 - Verwendung eines Multiplexers (z.B. Communication Gateway)
- Transfer des Konzeptes
 - Implementierung des Konzeptes für Rovermissionen

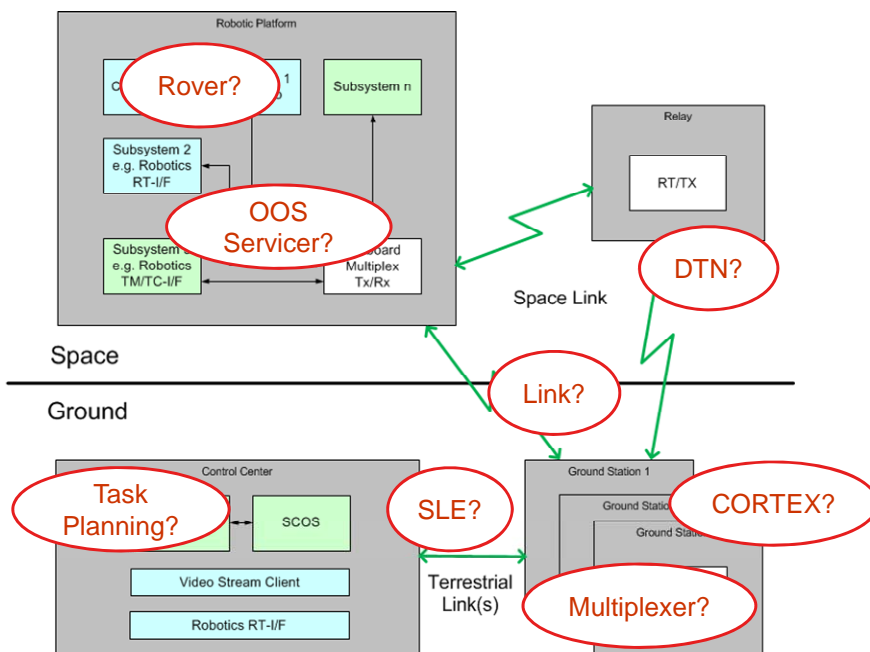


Abbildung 44: Verwendung des MICCRO-Konzeptes für andere Missionen mit robotischen Komponenten

3.3 Fortschritte auf dem Gebiet des Vorhabens

Fortschritte auf dem Gebiet des Vorhabens wurden insoweit erzielt, als dass die Studien zur Mission DEOS parallel weitergeführt wurden und hier sehr ähnliche Konzepte verfolgt werden.

Weitergehende Fortschritte auf eine harmonisierte robotische Schnittstelle wie sie von der CCSDS Working Group vorangetrieben wird, sind noch nicht implementiert.

3.4 Veröffentlichungen der Projektergebnisse

Die Ergebnisse wurden genutzt, um in Treffen mit potentiellen Partnern für eine wirtschaftliche Verwertung Aufmerksamkeit zu erzeugen.

Zusätzlich zu den reinen Projektmeetings wurden durch den Besuch von Fachveranstaltungen und entsprechenden Vorträgen und Veröffentlichungen die erzielten Resultate bekanntgemacht:

- iSAIRAS 2010, Sapporo (Japan) 2010-08-29 bis 2010-09-01, Teilnehmer: Ulli Leibnitz und Heiko Müller
- Besuch und Vortrag ASTRA 2011 Noordwijk, The Netherlands April 12-14, 2011:
 - Vortragstitel „Mission Control Concepts for Robotic Operations“
- Besuch und Vortrag European Ground System Architecture Workshop (ESAW) 2011, ESOC, Darmstadt, Germany, 10-11 May 2011:
 - Vortragstitel „ Ground Segment Autonomy: A Revised Approach “, Teilnehmer: Ulli Leibnitz, Heiko Müller, Markus Plura
- Besuch und Vortrag IAC 2011 Kapstadt, April 02.-07.10.2011:
 - Vortragstitel „ Mission Operations Concepts for Robotic Missions“
- Besuch und Vortrag 2. Nationale Konferenz zur Raumfahrt-Robotik, Berlin, Konferenzzentrum des BMWI:
 - Dr. Markus Plura
 - „Missionskontrollkonzepte für Weltraumrobotik (MIKKRO)“
 - 6.-7.03.2012
- Besuch und Vortrag SpaceOps 2012, Stockholm, Schweden:
 - Dr. Markus Plura, Martin Stelzer, Andreas Ohndorf
 - „MISSION CONTROL CONCEPTS FOR ROBOTIC OPERATIONS (MICCRO)“
 - 11.-15.06.2012
- Besuch und Vortrag iSAIRAS 2012, Turin, Italien:
 - Dr. Markus Plura
 - „END-TO-END CONCEPT DEMONSTRATION FOR ON-ORBIT SERVICING TYPE MISSIONS“
 - 04.-06.09.2012

3.5 Wichtigsten Positionen des zahlenmäßigen Nachweises

Im Projekt MICCRO waren die wesentlichen Kosten die Personalkosten. Rechnerisch sind die Kosten von den Personalstellen der Projektpartner abgerechnet worden.

3.6 Notwendigkeit und Angemessenheit der geleisteten Arbeit

Der Verlauf der Arbeit im Projekt folgte der im Projektantrag formulierten Planung.

Alle im Arbeitsplan formulierten Aufgaben wurden erfolgreich bearbeitet, es waren keine zusätzlichen Ressourcen für das Projekt nötig. Innerhalb des Projektes wurden erforderliche Ressourcen nach Bedarf bereitgestellt.



4 Erfolgskontrollbericht

Der Erfolgskontrollbericht zum Schlussbericht liegt diesem Dokument als Anlage 1 bei. Der Erfolgskontrollbericht wird nicht veröffentlicht.

5 Kurzfassung

Eine Kurzfassung (Berichtsblatt) des Schlussberichtes liegt diesem Dokument als Anlage 2 bei.

Berichtsblatt

1. ISBN oder ISSN	2. Berichtsart Schlussbericht	
3a. Titel des Berichts Schlussbericht (per 31.12.2012) nach Anlage 2 zu Nr. 8.2 NKBF 98 "Mission Control Concepts for Robotic Operations"		
3b. Titel der Publikation		
4a. Autoren des Berichts (Name, Vorname(n)) Dr. Markus Plura, Martin Stelzer, Andreas Ohndorf	5. Abschlussdatum des Vorhabens 30.09.2012	
	6. Veröffentlichungsdatum	
4b. Autoren der Publikation (Name, Vorname(n))	7. Form der Publikation	
8. Durchführende Institution(en) (Name, Adresse) SCISYS Deutschland GmbH Borgmannstrasse 2 44894 Bochum Institut für Robotik und Mechatronik Münchener Straße 20 82234 Weßling Deutsches Raumfahrt-Kontrollzentrum (GSOC) Münchener Straße 20 82234 Weßling	9. Ber.Nr. Durchführende Institution VCS-MICCRO-MGT-PR-00049	
	10. Förderkennzeichen 50 RA 1016	
13. Fördernde Institution (Name, Adresse) Raumfahrtmanagement des Deutschen Zentrums für Luft- und Raumfahrt e.V. Königswinterer Str. 522-524 53227 Bonn	11a. Seitenzahl Bericht 57	
	11b. Seitenzahl Publikation	
	12. Literaturangaben	
	14. Tabellen 7	
16. Zusätzliche Angaben	15. Abbildungen 44	
17. Vorgelegt bei (Titel, Ort, Datum) iSAIRAS 2012, Turin, Italien; SpaceOps 2012, Stockholm, Schweden; 2. Nationale Konferenz zur Raumfahrt-Robotik 2012, Berlin (vgl. Kapitel 3.4 Schlussbericht)		
18. Kurzfassung Innerhalb des Vorhabens wurde von den Projektpartnern SCISYS Deutschland GmbH (vormals VCS AG), DLR-RMC und DLR-GSOC eine Analyse der Betriebskonzepte vergangener und zukünftiger Missionen im Hinblick auf Gemeinsamkeiten durchgeführt. Hierbei wurde in der ersten Projektphase ein repräsentatives Missionskontrollkonzept für robotische Missionen erarbeitet, welches die bestehenden Betriebskonzepte, Verantwortlichkeiten, Informationsflüsse und Missionsphasen einbezieht, bzw. berücksichtigt. Diese Ergebnisse wurden in mehreren Veröffentlichungen verarbeitet und in Dokumenten nach ECSS-E-ST-40C, technischen Notizen und im Rahmen des Berichtswesens festgehalten. Neben der konzeptuellen Analyse wurde ein Demonstrationsprototyp implementiert, der eine Überprüfung des entwickelten Missionsbetriebskonzepts an einem repräsentativen System incl. eines simulierten Raumsegmentes zulässt. Das Design und der aktuelle Integrationsstand des am DLR Standort Oberpfaffenhofen implementierten Systemdemonstrators wurden in einem Vortrag erläutert. Durch die Verwendung dieses Systems konnten Aspekte wie die Übergabe zwischen verschiedenen Missionsphasen und deren Auswirkungen auf Rollen und Verantwortlichkeiten untersucht werden. Besonders für den Fall der telepräsenten Steuerung des Roboters werden neue Anforderungen in Bezug auf die Kommunikationsarchitektur aber auch die Prozesse zur Entscheidungsfindung gestellt. Die Projektergebnisse liefern hier entsprechende Lösungsansätze für zukünftige Missionen und können somit direkt in aktuellen Raumfahrtmissionen wie etwa DEOS eingebracht werden.		
19. Schlagwörter On-Orbit Servicing, Missionskontrollsystem, Telerobotik, Autonomie		
20. Verlag	21. Preis	

Document Control Sheet

1. ISBN or ISSN	2. Type of Report Final Report	
3a. Report Title Schlußbericht (per 31.12.2012) nach Anlage 2 zu Nr. 8.2 NKBF 98 "Mission Control Concepts for Robotic Operations"		
3b. Title of Publication		
4a. Author(s) of the Report (Family Name, First Name(s)) Dr. Markus Plura, Martin Stelzer, Andreas Ohndorf		5. End of Project 30.09.2012
4b. Author(s) of the Publication (Family Name, First Name(s))		6. Publication Date
8. Performing Organization(s) (Name, Address) SCISYS Deutschland GmbH Borgmannstrasse 2 44894 Bochum Institut für Robotik und Mechatronik Münchener Straße 20 82234 Weßling Deutsches Raumfahrt-Kontrollzentrum (GSOC) Münchener Straße 20 82234 Weßling		7. Form of Publication
13. Sponsoring Agency (Name, Address) Raumfahrtmanagement des Deutschen Zentrums für Luft- und Raumfahrt e.V. Königswinterer Str. 522-524 53227 Bonn		9. Originator's Report No. VCS-MICCRO-MGT-PR-00049
		10. Reference No. 50 RA 1016
		11a. No. of Pages Report 57
		11b. No. of Pages Publication
		12. No. of References
		14. No. of Tables 7
		15. No. of Figures 44
16. Supplementary Notes		
17. Presented at (Title, Place, Date) iSAIRAS 2012, Turin, Italien; SpaceOps 2012, Stockholm, Schweden; 2. Nationale Konferenz zur Raumfahrt-Robotik 2012, Berlin (cp. chapter 3.4 final report)		
18. Abstract Within the project control concepts of past und future space missions have been analyzed by the project team regarding their commonalities. Based on this analysis a mission control concept for a robotic space mission was created in the first project phase, which considers existing concepts for operation, responsibility, information flow and mission phases. The results have been published on various conferences and in the internal project documentation according to ECSS-E-ST-40C. To verify this mission control concept in a representative environment a demonstrator including a simulated space segment was implemented in the second project phase. Aspects such as handshakes between different mission phases and their influence on roles and responsibilities were investigated using this setup. Especially for the tele-presence mode new requirements arise for the communication architecture and also for the process of finding a decision. The results of the project provide solutions for this matter for future space missions and can, therefore, be directly used in projects such as DEOS.		
19. Keywords On-Orbit Servicing, Mission Control System, Telerobotics, Autonomy		
20. Publisher		21. Price