SET-UP AND VALIDATION OF METERON END-TO-END NETWORK FOR ROBOTIC EXPERIMENTS.

Francois Bosquillon de Frescheville (1) , Sebastian Martin (2), Nicola Policella (3), David Patterson (4) , Manuel Aiple (5) , Paul Steele (6) .

(1) ESA/ESOC, Robert-Bosch-Strasse 5, 64293 Darmstadt, Germany, francois.bosquillon.de.frescheville@esa.int
(2) Solenix Deutschland GmbH, c/o ESA/ESOC, Robert-Bosch-Strasse 5, 64293 Darmstadt, Germany, sebastian.martin@esa.int
(3) Solenix Deutschland GmbH, c/o ESA/ESOC, Robert-Bosch-Strasse 5, 64293 Darmstadt, Germany, nicola.policella@esa.int
(4) VEGA Space GmbH, Europaplatz 5, 64293 Darmstadt, Germany, c/o ESA/ESOC, david.patterson@esa.int
(5) ESA/ESOC, Robert-Bosch-Strasse 5, 64293 Darmstadt, Germany, manuel.aiple@esa.int
(6) ESA/ESOC, Robert-Bosch-Strasse 5, 64293 Darmstadt, Germany, paul.steele@esa.int

ABSTRACT

This paper provides a short presentation of the Multi-purpose End-To-End Robotic Operations Network (METERON). This project aims at simulating selected future Human exploration scenarios including immersive remote control of a robot by an astronaut in orbit around a target object (such as Mars or the Moon). The remote human orbiter is simulated by the International Space Station (ISS). The robot(s)/rover(s) will be located on a simulated planetary surface located at a site on Earth. The ground segment is split into two independent parts: The mission operations infrastructure and the emulation of the robot communications. The authors are focusing on the operations and communications elements of METERON and will explain how the end-to-end infrastructure is being set–up and will be tested. The non-real time operations and communications chains between the ground and ISS are being established in cooperation with NASA. The current baseline for the real-time control links between the ISS operator and the ground robot is to use the KONTUR-2 communications system in collaboration with Roskosmos and DLR. This involves the set-up of communications paths on-board the ISS, providing communications between the Russian, American, and European modules. This paper focuses on Phase 1 of the METERON project for which the robotics communication data generator and receiver will be simulated using a simple robot Simulator: MOCUP (METERON Operations and Communications Prototype) controlled through MOPS (MOCUP Operations Software). The current plan calls for an end-to-end demonstration of the operations and communications elements of METERON towards the end of 2011/beginning 2012.

From this point on it is assumed that the communications baseline will be

1. **Non-Real time communications via the NASA DTN system**
2. **Real time communications via the Russian KONTUR-2 system**

Both of these assumptions are pending formal agreements with the respective agencies/institutes.

1. INTRODUCTION

Project METERON (the Multipurpose End-To-End Robotic Operations Network) was initiated in response to a call to start using the ISS as a test bed for future space human exploration made at the IAC congress in Glasgow (2008) in a plenary session.

Two and a half years later, METERON [1] is proposed by the ESA Human Spaceflight and Operations (D/HSO) and Technical and Quality Management (D/TEC) directorates as an international collaboration between ESA, NASA (University of Colorado), Roskosmos and DLR. It intends to use the ISS as a test bed to simulate an orbiter around another heavenly body (for example Mars), under directives from Mission Control on Earth. Astronauts on the orbiter will project their human initiative and instinct, in *real-time,* onto the surface of the heavenly body (simulated by an analog site on the Earth) through robotic device(s) to perform science or engineering tasks. This type of real-time control is not possible directly from Earth due to the One Way Light Time delay in communications.

It is likely that any future human exploration strategy will at least have a phase where such orbit to surface remote activities will need to be performed. This is confirmed independently by studies reported by NASA papers (the flex path rationale [2], and the HERRO concept [3]). Manned orbiter missions (without astronauts landing on the surface) strongly reduce the costs and risks associated with landing humans on a remote body or planet.

A test bed such as METERON provides an in-orbit test-bed to the robotic community in the areas of the robotic control in flight and the robotic assets used on the ground, as well as for the communications and operations aspects. The ISS is a highly complex space environment, and is therefore representative of orbiters considered for future human exploration missions.

METERON experiments will generate data and lessons learned that will be useful for the advancement of robotics, operations, communications and ergonomics in the context of human space exploration.

The following sections describe the process being followed to set-up and test the end-to-end communications chain concluded by an Operational Readiness Demonstration.

2. SHORT DESCRIPTION OF THE METERON ARCHITECTURE

In order to understand what needs to be tested, a short description of the METERON infrastructure is presented hereafter and shown in Fig.1 (the figure is reproduced from [1]):

<u>Earth to manned orbiter (ISS):</u> the baseline foresees non-real-time communications via the NASA DEN (DTN (Delay-Tolerant Networking) Experiment Network)) which already extends up to the ISS with a 100bps uplink and a 500 Kbits downlink. Alternative routes, such as via the Columbus payload chain, are also possible.

<u>Manned orbiter:</u> a laptop located in the Columbus module will be connected to the DEN on one side and to the KONTUR-2 (previously called ROKVISS [4]) real time low latency system on the other. It will allow communications either independently or across these two links.

<u>Manned orbiter to surface robotic element:</u> a network of ground stations (3) acquires the KONTUR-2 signal in succession in order to simulate a 15 minutes duration visibility. The stations are connected via land lines to the location of the robotic element(s).

In addition, the architecture is controlled and monitored from the METERON control centre. The control centre has a real time or near real time overview of all actors, equipments and events occurring. It uses this information to simulate operation scenarios.

This infrastructure is developed in two phases: Phase 1 is validating the concept with simple, low cost and non redundant equipment while phase 2 will see the total deployment of the infrastructure.

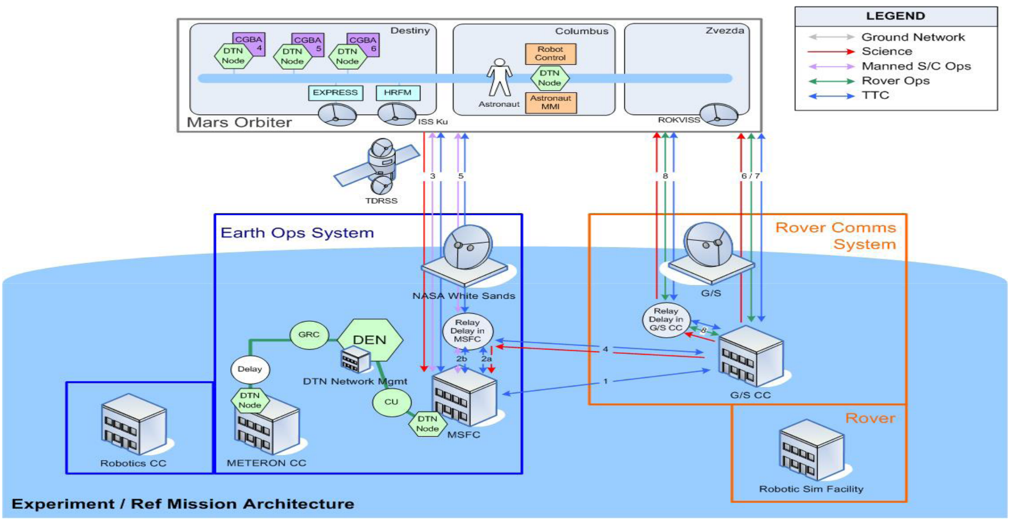

Figure 1: METERON infrastructure

3. METERON PHASE ONE DETAILED ACTIVITIES

To marry the two complementary operational concepts of METERON, real-time and non-real-time, Phase 1, also known as QuickStart, foresees three steps:

QuickStart-a. Establish basic DTN connectivity between ESOC and the Columbus module on the ISS in collaboration with the University of Colorado. The scope is to use DTN protocols to allow a control facility in the European Space Operations Centre (ESOC) to interact with the laptop on the Columbus module on the ISS.

QuickStart-b. Enable basic real-time special-purpose telerobotics communication between the Columbus module and the ground, using the Russian KONTUR-2 direct link. The major goal will be to establish the onboard connectivity between the two ISS modules.

QuickStart-c. Demonstrate full end-to-end connectivity between ESOC and a ground telerobotics test facility (via the ISS and KONTUR-2), augmented by special-purpose direct telerobotics connectivity between the Columbus module and the test facility, using the KONTUR-2 link. This phase will integrate the results of the two previous phases.

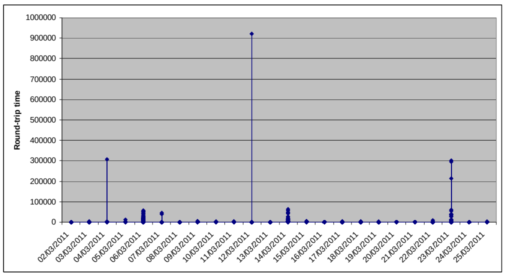

Figure 2: Performance results

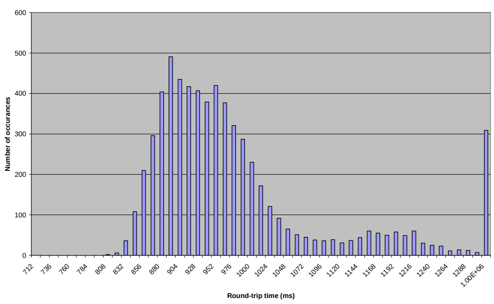

Figure 3: Round-trip frequency over 6410 samples

Fig.2 and Fig.3 show the performance results obtained from an intensive evaluation test of the communication link between ESOC and the University of Colorado performed during QuickStart-a. The test consisted of 6410 samples distributed over 23 consecutive days. The results show that the DTN link is quite stable with over 95% of the samples with a round-trip time below 1,5s. In order to have a robust and controlled communication link, future work will analyze the few cases of failure (less than 0.8% of the sample) or of longer round-trip time.

The real time control links between the ISS operator and the ground robot are being established through collaboration with Roskosmos, the Central R&D Institute of Robotics and Technical Cybernetics in St Petersburg and DLR. This involves the set-up of communications paths on-board the ISS, linking the Russian, American, and European modules, as well as ground communication between the ground station(s) and the robotic centre.

To conclude METERON phase one an **Operational Readiness Demonstration** will be conducted. The connectivity capabilities developed in the different steps will be demonstrated with the support of a purpose-built robotic prototype which will emulate the robotic systems on the ground.

4. MOCUP AND MOPS

MOCUP (METERON Operations and CommUnications Prototype) is a simplified prototype of a basic exploration rover designed for communications testing. The overall requirement was to be able to simulate the end-to-end communication chain with non-real time and real-time channels and to test operating concepts in the METERON context. MOCUP was, therefore, designed to be able to drive around in an easy environment so that it could be tele-operated in real-time and in sequential mode, capturing its environment with a camera.

MOPS (MOCUP Operations Software) is a suite of software tools for controlling MOCUP. It consists, in particular, of a graphical user interface (MOPSUI) running on the operator's machine and of controller software running on the robot's embedded computer.

Thus, MOCUP allows execution of end-to-end experiments for METERON before full-scale robotics hardware is available, helping to demonstrate operational concepts and to test communication chains at a level beyond a software simulator.

The development of MOCUP explicitly took place following a rapid prototyping approach, prioritizing simplicity and short development time over robustness and performance. It was built using the LEGO® MINDSTORM NXT kit combined with an off-the-shelf embedded computer called Beagleboard. The architecture is shown in Fig.4.

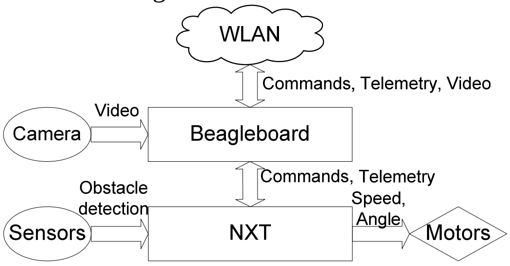

Figure 4: MOCUP architecture

LEGO MINDSTORM NXT comes with a controller to which three electrical motors and four sensors can be connected. For programming the NXT controller, firmware was replaced by the leJOS firmware, a Java Micro Edition virtual machine which contains a set of Java classes for the NXT. This simplifies the reading of sensor data and the control of the motors.

MOCUP is 42cm long, 29cm wide and 24cm high. A caterpillar design was chosen in which two of the available motors are driving the left and right tracks, the third motor drives the camera inclination system see Fig.5 and Fig.6.

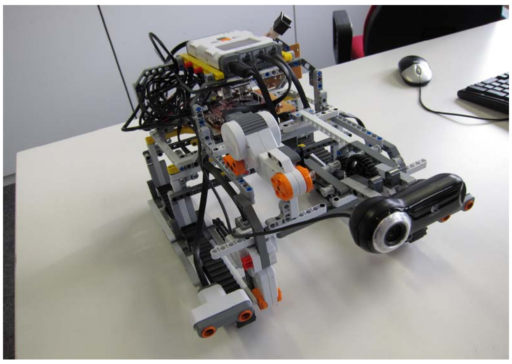

Figure 5: MOCUP view from the front

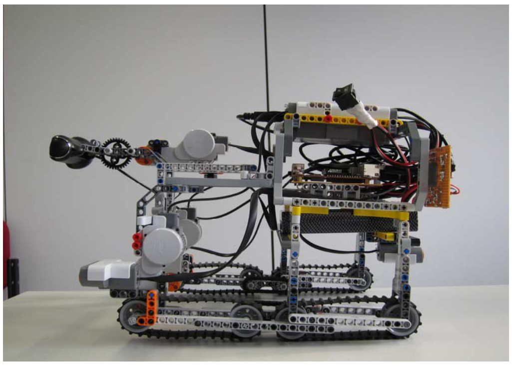

Figure 6: MOCUP view from the left side

Three ultrasonic distance sensors are used on-board for obstacle detection: two at the front (left and right side) and one at the rear (centre). This enables basic protection against self-induced damage, especially in sequence mode.

While the NXT controls the robotic hardware, the Beagleboard embedded computer runs the communication stack and the video feedback: it is equipped with a Wireless LAN adapter through which the link to the METERON communication infrastructure is established. The standard Linux UDP over IP stack is used for real-time communication whereas a DTN node running on the Beagleboard covers the non real-time communication. A USB camera provides video data, which is compressed on the Beagleboard and streamed through a real time protocol channel when requested.

The Beagleboard and the NXT are linked through a USB connection. Software running on the Beagleboard manages the communication between MOCUP and the METERON communication infrastructure, i.e. it receives commands over DTN/UDP and forwards them to the NXT or forwards the telemetry from the NXT to the DTN/UDP.

The operator can control MOCUP using MOPSUI application as shown in Fig.7.

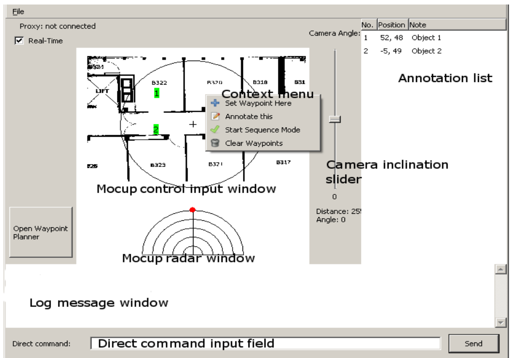

Figure 7: MOPSUI screenshot

The control input window displays the position and track of MOCUP on a map of the environment. It also includes positions of obstacles detected by the sensors.

The operator can steer MOCUP directly when operated in real-time mode using an input similar to a joystick, which captures the desired velocity vector. In addition the operator can also modify the camera inclination angle using the camera inclination slider.

In non real-time mode, the operator can set a list of waypoints on the map and then start a sequence or load a list of commands from a file. The operator can directly interact with the map using the context menu to set waypoints.

To simplify the generation of exploration reports, the operator can also annotate points on the map using the annotation list. These annotations can then be exported together with the track of MOCUP and the detected obstacles to a PDF file as an overlay of the original map.

In a separate window the operator sees the video stream from the MOCUP onboard camera (in real-time mode).

As pointed out earlier, the real-time performance of the overall system is not the one expected of future robotic systems, which may include other actuators. The measured delay is around 25-50ms end-to-end. For the task of driving MOCUP in real-time, however, this delay is small enough. In tests, following a brief introduction, it turned out to be intuitive and easy to steer, even without direct visibility of the rover.

5. THE END-TO-END NETWORK SET-UP AND TESTING

As previously introduced, the METERON network has a non-real-time, DTN defined segment at one end and a real-time, Quality-of-Service defined segment at the other (see Fig.8).

The ESOC DTN nodes are virtual machines running on a server hosted on the ESOC external LAN and connect via VPN over a standard Internet connection to the NASA DEN (which is responsible for the link to the ISS via TDRSS). Section 6.1 describes the setup and testing of this segment.

On board the ISS, the 100bps non-real-time link is complemented by a 4Mbps down / 256kbps up real-time link for the purpose of force-feedback control of a robot on the ground by an astronaut in orbit. At the hub of this connection the current assumption is that there will be a single laptop in the Columbus module, which will connect the Columbus LAN with the Zvezda LAN.

The laptop will serve a number of purposes:

- Bridge from the Bundle Protocol of the DTN link to the UDP/IP protocol for transmission across the LAN to the KONTUR-2 On-Board Computer and vice versa,
- Bridge from the Bundle Protocol of the DTN link to the TCP/IP protocol for a Monitoring & Control (M&C) interface to the robot controller itself,
- Host an M&C client for use by the astronaut.

The robot controller will have a ‘direct’ (not via the laptop) UDP/IP link to the KONTUR-2 OBC for the purposes of robot monitoring and control. The OBC, in turn, will be responsible for bridging between the UDP and CCSDS Space Packet protocols.

The DLR Weilheim station used for the ROKVISS experiments has an already proven interface to KONTUR-2. In addition, for the short-term QuickStart demonstrations, the 3.5m Redu-3 ground station is being prepared to enable communications with the KONTUR-2 module.

It is beyond the scope of this paper but, to complete the picture of an exploration mission in orbit around a planet or an asteroid, the ‘astronaut chain’ is also considered part of the METERON network. This will involve the Columbus Control Centre which will ‘manage the astronaut’ incorporating the robot experiments into their daily schedules and protocols.

Note that anything to do with actual robotic experiments is part of METERON Phase 2. The elements of Phase 1 covered by this paper are concerned only with communications demonstration, validation and characterisation.

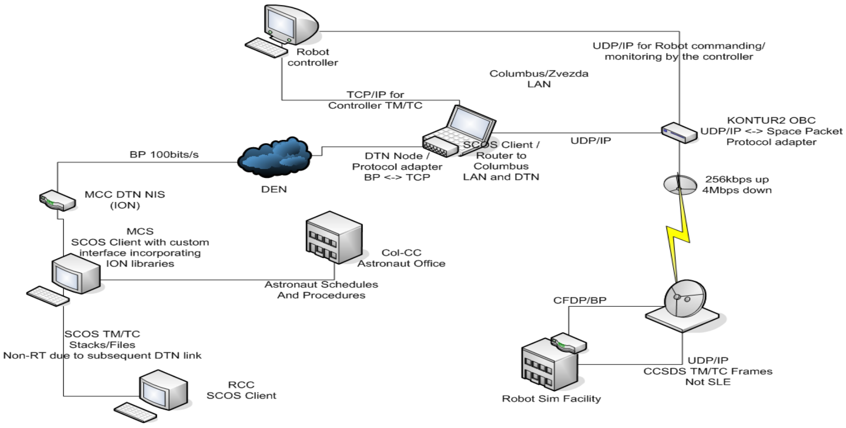

Figure 8: METERON communication Network

More relevant to Phase 2, but again for information, there will be an Out-of-Band network connecting all the METERON elements to the METERON Simulation Facility. This will effectively be a 'simulation M&C' overlay on top of the 'operations' network. The Simulations Officers will be able to use this network to manipulate the events of the planetary mission scenarios that can be played out using METERON.

6. COMMUNICATION ASPECTS

6.1 Non-real-time links

DTN is a networking technology, which enables communication through a network subject to very long signal propagation latency and/or prolonged intervals of unavailability. In particular, the network can be a network of regional networks (internets). The DTN then represents a layer or overlay on top of these networks including the Internet.

DTNs are characterized by their lack of connectivity, resulting in a lack of instantaneous end-to-end paths. DTNs overcome these problems by using store-and-forward message switching. This functionality is implemented by overlaying a new protocol layer – the bundle layer. This layer ties together the region-specific lower layer so that application programs can communicate across multiple regions.

A DTN node is a network entity with a bundle layer. The node might have the possibility of sending, receiving, and/or forwarding bundles (functionality will be allocated depending on the role of the node, i.e. host, router, gateway).

These DTN features, in combination with pre-definable communication windows and communication opportunities, provide a much more suitable protocol for space applications than standard IP protocols. The Interplanetary Overlay Network (ION) [5] software distribution is one possible implementation of the Delay-Tolerant Networking (DTN) architecture as described in Internet RFC 4838 [6], and has been chosen for the METERON non-real-time communications link baseline. ION is designed to enable inexpensive insertion of DTN functionality into embedded systems such as robots or spacecraft.

For fast prototyping and to facilitate first developments, the communication between MOCUP and MOPS evolved in iterative steps from standard TCP/IP based communication to DTN bundle protocol based communication. This allowed the creation and validation of the first command & controlling software functionality in a familiar environment before adding the complexity of new protocols. The next step was to establish DTN proxies on the MOCUP and MOPS controlling machines to act as tunnel for DTN communication. The last step was to deploy DTN support in the control software and on the MOCUP hardware itself thus making the proxies for translating from TCP/IP to bundle protocol and back obsolete.

Once the links had been established and verified, additional nodes could easily be added by extending the DTN configuration with additional hops from the ESOC DTN nodes to the NASA DTN gateway and beyond. This gradual inclusion of nodes helped to measure the effects and delays of extending the links and to successively introduce necessary firewall and security principles.

Through these tests, important aspects of the non-real-time communications could already be tested and validated early on in the project. The results gathered in small successive steps will be very interesting and useful for the next phases of the project.

6.2 Real-time links

Tele-presence operation of a robot with force-feedback and video has exacting and uncommon requirements on the communication link path:

- High (Mbps) data rates. On the downlink this is nothing special in space missions. For METERON, 4Mbps uplink would be a representative data rate to allow for video feedback (as already used in the ROKVISS experiment on the downlink) but due to regulations concerning radiating to space in general, and to the ISS in particular, this is limited to 256kbps.
- 100ms round trip is the maximum allowable delay. During QuickStart there will be no deterministic robotic equipment on the ISS so the Columbus laptop will have to serve as the end point for these tests. Its OS will be Ubuntu 10.04. This is not a real-time OS but the 2.6 linux kernel is arguably responsive enough that, with a large enough set of ping tests, a reasonably good impression of the link's latency can be established. Alternatively, the *pre-empt* or *low-latency* version of the kernel can be used. A suitable Quality-of-Service

(QoS) link between the ground station and the robot facility will be rented.

- 2ms is the maximum allowable jitter. This is also a requirement on the QoS link on ground. Any margin in the latency will be used to smooth out variance in the rate of reception. In general, the real-time link between the robot and its controller must be as interruption-free as possible (minimise the amount of equipment in the path).

6.3 Custom Protocols

Apart from the communication layer aspects, application layer aspects also had to be considered. Although the first rudimentary MOCUP commanding protocol was created simply to be issued from a command line and is therefore, text based, a more PUS or database supported protocol needs to be established for the future. This is required to enable better handling of the commanding via applications, provide traceability of the current state via sequence numbers and acknowledgements and to be compatible with and allow reusability of available ESA infrastructure software such as test environments, simulators and control systems. Standardized protocols will in addition allow plug-and-play of different robotic controllers and robots or rovers.

7. TEST ENVIRONMENT

To prepare for operations and to prove the concepts, the test environment needs to satisfy the following requirements:

i. Apply a communications standard between the METERON sites,
ii. Be deployable and controllable from various sites in isolation or in an integrated simulation,
iii. Be scalable.

A simulation environment such as ESA's Ground Segment Test and Validation Infrastructure (GSTVi) [7] combines several existing simulators already used to test and validate operations for ESA spacecraft. Being modular, the simulations infrastructure will be used to demonstrate and validate inter-site communication and protocols and then to gradually replace the simulated component (Operations Centre/ Ground station/ Orbiter/ Rover) with the real hardware in the loop. A PUS standard communication level is implemented and further expansion is planned to investigate and possibly incorporate DTN, AMS and SM&C communication.

Using a simulator allows one to oversee all levels of communication, either human, HW or SW. Failures can be inserted in individual units, interfaces, bit corruptions, biases, late verifications etc. and the end effect of these failures can be noted on the efficiency and safety of the executed operations. Lessons learned will then feed back into improving the network (or recommending standards) and ensuring the highest quality operations at the least overhead to the sites and personnel.

In Phase 1, real-time remote monitoring of the test robotic experiments (with MOCUP) via simulator displays at a central site will be demonstrated, using a system as illustrated in Fig.9. Once the format of commands and layers is standardised, remote control of a robotic experiment can be attempted. To this end, GSTVi is being customised to allow visualisation of operational data (HK) from e.g. the robotic experiments. Deploying DTN within GSTVi is also ongoing although it has been demonstrated in a separate ESA study.

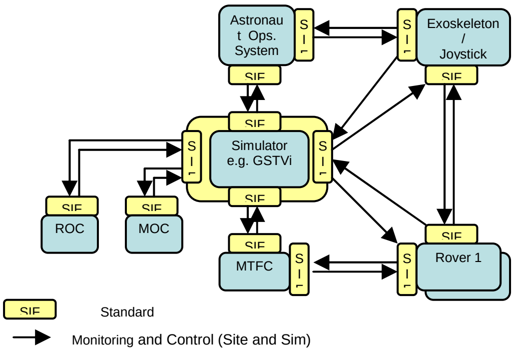

Figure 9: GSTVi METERON test environment

8. VALIDATION OF THE METERON END-TO-END CHAIN USING MOPS AND MOCUP

A series of the METERON Communications tests have been defined and executed using MOCUP and MOPS

In real-time mode MOCUP was successfully controlled over a standard LAN environment using MOPS. In this test telecommands, telemetry and video payload data were transmitted. MOCUP could be controlled in real time to explore an unknown remote environment in an intuitive way. This test could be extended to test long distance connections with off-the shelf technology. The next test will be to bridge the connection over a space link. Work is currently in progress at DLR to implement interface software that will transmit UDP packets in CCSDS TM/TC frames, as part of the KONTUR-2 communications chain upgrade. This software will run on the KONTUR-2 onboard computer and the ground station terminal thus completing the real-time chain between ISS and the robotics facility.

Another test that has been completed was to control MOCUP over local DTN nodes using MOPS in non-real time mode. In this test only telecommands and telemetry and no video payload data were transmitted.

Independently, the three node end-to-end DTN connection between ESOC and CU Boulder was established and tested (two DTN nodes at ESOC communicating via the CU Boulder node only). This test provided important results relevant to configuration of DTN in the context of METERON. The next steps will be to connect to the NASA DTN network and test connections to the Commercial Generic Bioprocessing Apparatus (CGBA) [8] onboard ISS from ESOC.

9. CONCLUSION

This paper has described how the METERON End-to-End communication path is being set-up during Phase 1. So far the non real time transmission path has been proven up to a DTN node in the University of Colorado using the MOCUP commanded in sequence mode. The rest of the link up to the laptop in Columbus is under NASA responsibility. The real time control of MOCUP with video feedback was also proven in different local configurations. This different results are paving the way towards the Operational Readiness Demonstration at the end of 2011 or begin of 2012 using the robotic communications test system comprising of MOPS and MOCUP This will open the way for METERON Phase 2, which will entail full deployment of the ground segment, space component and robotic increment execution as well as operation scenarios simulations.

ACKNOWLEDGEMENTS:

To B. Chesson ESA OPS-H who has fully supported the apparently illogical idea to use an astronaut in ISS to control a humanoid or robot on ground from the first minute it was exposed to him.

To K. Nergaard, Ph. Schoonejans, A. Schiele, J. Grenouilleau, B. Carey, K. Landzettel, A. Hooke, E. J. Wyatt, K.F. Nichols, K. Gifford, K. Scott, M. Kearney, L. Pitts, S. Burleigh, S. Williams who are working with the authors on Quickstart-a and Phase 2 preparations.

REFERENCES:

1. METERON Concurrent Design Facility study report (available on request from kim.nergaard@esa.int), ESA CDF-96(A), Dec 2009.

2. G.R. Schmidt et al, Rationale for Flexible Path – A Space Exploration Strategy for the 21st Century, *Journal of British Interplanetary Society*, 63, pp.42-52, 2010.

3. Schmidt, G. R., Oleson, S. R., Borowski, S. K., and Landis, G. A., “HERRO: A Science-Oriented Strategy for Crewed Missions Beyond LEO,” paper AIAA-2010-0629, *AIAA Aerospace Sciences Conference*, January 6-9 2010, Orlando FL.

4. Preusche, C.; Reintsema, D.; Landzettel, K.; Hirzinger, G.; "Robotics Component Verification on ISS ROKVISS - Preliminary Results for Telepresence" Inst. of Robotics & Mechatronics, German Aerosp. Center, Wessling *Intelligent Robots and Systems, 2006 IEEE/RSJ International Conference on* 9-15 Oct. 2006 in Beijing pages 4595 - 4601

5. Interplanetary Overlay Network (ION) design and Operation, (JPL D-48259), V1.8, Jet Propulsion Laboratory 6 February 2008.

6. Request for Comments 4838, http://www.rfc.editor.org/rfc/rfc4838.txt

7. C. Laroque, M. Goetzelmann, M. Pecchioli, N. Sebastiao – A Virtual Space System for Ground Data Systems Validation, *SpaceOps 2008*.

8. K. Nichols, M. Holbrook, R. Lee Pitts, Kevin K. Gifford, A. Jenkins, and S. Kuzminsky. DTN Implementation and Utilization Options on the International Space Station. In Proceedings of SpaceOps 2010 Conference, 25 - 30 April 2010, Huntsville, Alabama, USA.