



The Consultative Committee for Space Data Systems

Report Concerning Space Data System Standards

TELEROBOTIC OPERATIONS

INFORMATIONAL REPORT

CCSDS 540.0-G-1

GREEN BOOK

November 2016

Report Concerning Space Data System Standards

**TELEROBOTIC
OPERATIONS**

INFORMATIONAL REPORT

CCSDS 540.0-G-1

GREEN BOOK

November 2016

AUTHORITY

Issue:	Informational Report, Issue 1
Date:	November 2016
Location:	Washington, DC, USA

This document has been approved for publication by the Management Council of the Consultative Committee for Space Data Systems (CCSDS) and reflects the consensus of technical panel experts from CCSDS Member Agencies. The procedure for review and authorization of CCSDS Reports is detailed in *Organization and Processes for the Consultative Committee for Space Data Systems* (CCSDS A02.1-Y-4).

This document is published and maintained by:

CCSDS Secretariat
National Aeronautics and Space Administration
Washington, DC, USA
E-mail: secretariat@mailman.ccsds.org

FOREWORD

Agencies agree that human space exploration will be most successful as an international endeavor, because there are many challenges to preparing for these missions, and because of the significant social, intellectual, and economic benefits to people on Earth.

– The Global Exploration Roadmap

Through the process of normal evolution, it is expected that expansion, deletion, or modification of this document may occur. This Report is therefore subject to CCSDS document management and change control procedures, which are defined in *Organization and Processes for the Consultative Committee for Space Data Systems* (CCSDS A02.1-Y-4). Current versions of CCSDS documents are maintained at the CCSDS Web site:

<http://www.ccsds.org/>

Questions relating to the contents or status of this document should be sent to the CCSDS Secretariat at the e-mail address indicated on page i.

At time of publication, the active Member and Observer Agencies of the CCSDS were:

Member Agencies

- Agenzia Spaziale Italiana (ASI)/Italy.
- Canadian Space Agency (CSA)/Canada.
- Centre National d'Etudes Spatiales (CNES)/France.
- China National Space Administration (CNSA)/People's Republic of China.
- Deutsches Zentrum für Luft- und Raumfahrt (DLR)/Germany.
- European Space Agency (ESA)/Europe.
- Federal Space Agency (FSA)/Russian Federation.
- Instituto Nacional de Pesquisas Espaciais (INPE)/Brazil.
- Japan Aerospace Exploration Agency (JAXA)/Japan.
- National Aeronautics and Space Administration (NASA)/USA.
- UK Space Agency/United Kingdom.

Observer Agencies

- Austrian Space Agency (ASA)/Austria.
- Belgian Federal Science Policy Office (BFSP0)/Belgium.
- Central Research Institute of Machine Building (TsNIIMash)/Russian Federation.
- China Satellite Launch and Tracking Control General, Beijing Institute of Tracking and Telecommunications Technology (CLTC/BITTT)/China.
- Chinese Academy of Sciences (CAS)/China.
- Chinese Academy of Space Technology (CAST)/China.
- Commonwealth Scientific and Industrial Research Organization (CSIRO)/Australia.
- Danish National Space Center (DNSC)/Denmark.
- Departamento de Ciência e Tecnologia Aeroespacial (DCTA)/Brazil.
- Electronics and Telecommunications Research Institute (ETRI)/Korea.
- European Organization for the Exploitation of Meteorological Satellites (EUMETSAT)/Europe.
- European Telecommunications Satellite Organization (EUTELSAT)/Europe.
- Geo-Informatics and Space Technology Development Agency (GISTDA)/Thailand.
- Hellenic National Space Committee (HNSC)/Greece.
- Indian Space Research Organization (ISRO)/India.
- Institute of Space Research (IKI)/Russian Federation.
- KFKI Research Institute for Particle & Nuclear Physics (KFKI)/Hungary.
- Korea Aerospace Research Institute (KARI)/Korea.
- Ministry of Communications (MOC)/Israel.
- National Institute of Information and Communications Technology (NICT)/Japan.
- National Oceanic and Atmospheric Administration (NOAA)/USA.
- National Space Agency of the Republic of Kazakhstan (NSARK)/Kazakhstan.
- National Space Organization (NSPO)/Chinese Taipei.
- Naval Center for Space Technology (NCST)/USA.
- Scientific and Technological Research Council of Turkey (TUBITAK)/Turkey.
- South African National Space Agency (SANSA)/Republic of South Africa.
- Space and Upper Atmosphere Research Commission (SUPARCO)/Pakistan.
- Swedish Space Corporation (SSC)/Sweden.
- Swiss Space Office (SSO)/Switzerland.
- United States Geological Survey (USGS)/USA.

DOCUMENT CONTROL

Document	Title	Date	Status
CCSDS 540.0-G-1	Telerobotic Operations, Informational Report, Issue 1	November 2016	Original issue

CONTENTS

<u>Section</u>	<u>Page</u>
1 INTRODUCTION.....	1-1
1.1 PURPOSE AND SCOPE.....	1-1
1.2 DOCUMENT STRUCTURE	1-1
1.3 DEFINITIONS, NOMENCLATURE, AND CONVENTIONS	1-1
1.4 REFERENCES	1-2
2 INTRODUCTION TO TELEROBOTIC OPERATIONS.....	2-1
2.1 OVERVIEW	2-1
2.2 SUMMARY OF TELEROBOTIC OPERATIONS.....	2-1
2.3 OVERVIEW OF TELEROBOTIC OPERATIONS LAYERS	2-2
2.4 POTENTIAL BENEFITS OF AN INTERNATIONAL STANDARD.....	2-3
2.5 RELATIONSHIP TO THE MISSION OPERATIONS SERVICES CONCEPT ..	2-4
3 DEFINITION OF TELEROBOTIC OPERATIONS.....	3-1
3.1 INTRODUCTION	3-1
3.2 TELEROBOTIC OPERATIONS SERVICES	3-1
3.3 TELEROBOTIC OPERATIONS SERVICES	3-9
3.4 OPEN QUESTIONS REGARDING TELEROBOTIC OPERATIONS	3-19
4 OVERVIEW OF THE MISSION OPERATIONS SERVICE FRAMEWORK.....	4-1
4.1 MISSION OPERATIONS SERVICE INTRODUCTION	4-1
4.2 MISSION OPERATIONS SERVICES CONCEPT	4-1
4.3 MISSION OPERATIONS SERVICES MESSAGE ABSTRACTION LAYER ...	4-2
4.4 MISSION OPERATIONS SERVICES COMMON OBJECT MODEL.....	4-3
4.5 MISSION OPERATIONS SERVICES COMMON SERVICES	4-4
4.6 MISSION OPERATIONS SERVICES MONITOR AND CONTROL	4-6
5 DOCUMENT ROADMAP.....	5-1
5.1 INTRODUCTION	5-1
5.2 TESTING AND PROTOTYPING	5-1
5.3 ELIMINATING OVERLAP WITH OTHER STANDARDS.....	5-1
5.4 SELECTION OF REFERENCE IMPLEMENTATION LANGUAGE.....	5-1

CONTENTS (continued)

<u>Section</u>	<u>Page</u>
6 OTHER INDUSTRY STANDARDS	6-1
6.1 OVERVIEW	6-1
6.2 JOINT ARCHITECTURE FOR UNMANNED SYSTEMS.....	6-1
6.3 ROBOT APPLICATION PROGRAMMING INTERFACE DELEGATE	6-1
6.4 COUPLED-LAYER ARCHITECTURE FOR ROBOTIC AUTONOMY	6-2
6.5 SENSOR MODEL LANGUAGE.....	6-3
6.6 ROBOT OPERATING SYSTEM	6-3
ANNEX A LEXICON	A-1
ANNEX B ABBREVIATIONS	B-1

Figure

2-1 Telerobotic End-to-End Model.....	2-1
2-2 A Strawman Gap Analysis.....	2-3
3-1 Artist’s Concept of Orbiter, Blimps, Rovers, and Robots Working Together	3-6
3-2 ESA Rio Tinto Field Test in Andalusia, Spain.....	3-8

1 INTRODUCTION

1.1 PURPOSE AND SCOPE

This CCSDS Informational Report contains overview and descriptive material supporting analysis, requirements, descriptions of use, scenarios, etc., that will help bound the scope of a proposed CCSDS Recommended Standard for telerobotic operations.

The contents of this Report are limited, generally, to information that adds value to the process of developing a proposed Recommended Standard for telerobotic operations. The main content of the Report is technical in nature. The following classes of non-technical data are specifically and purposely not included in this document:

- **Administrative:** This Report does not include ‘management’ or ‘collaboration’ information.
- **Advocacy:** Although additional materials describing current technical developments in collaborative telerobotics and standards are valuable, they are not included in this Report.

The information presented in this Report relates to the development of standards that support the safe, collaborative operation of mixed teams of human and robotic assets in the exploration of space.

1.2 DOCUMENT STRUCTURE

Following the introductory material, this document is organized into these main sections:

- Section 2: Introduction to Telerobotic Operations;
- Section 3: Definition of Telerobotic Operations;
- Section 4: Overview of the Mission Operations Service Framework;
- Section 5: Document Roadmap;
- Section 6: Other Industry Standards.

1.3 DEFINITIONS, NOMENCLATURE, AND CONVENTIONS

This Informational Report makes use of a number of terms defined in Mission Operations Services Concept (reference [3]). The use of those terms in this Informational Report is to be understood in a generic sense, i.e., in the sense that those terms are generally applicable to technologies that provide for the telerobotic operation of real systems. Those terms are:

- action;
- activity;

- alert;
- application programmers' interface;
- capability;
- domain;
- operation;
- service;
- service consumer;
- service provider.

Additional terms of reference and their definitions may be found in annex A, Lexicon.

1.4 REFERENCES

The following documents are referenced in this Report. At the time of publication, the editions indicated were valid. All documents are subject to revision, and users of this Report are encouraged to investigate the possibility of applying the most recent editions of the documents indicated below. The CCSDS Secretariat maintains a register of currently valid CCSDS documents.

- [1] *Mission Operations Message Abstraction Layer*. Issue 2. Recommendation for Space Data System Standards (Blue Book), CCSDS 521.0-B-2. Washington, D.C.: CCSDS, March 2013.
- [2] *The Global Exploration Roadmap*. International Space Exploration Coordination Group. NP-2013-06-945-HQ. Washington, D.C.: NASA Headquarters, August 2013.
- [3] *Mission Operations Services Concept*. Issue 3. Report Concerning Space Data System Standards (Green Book), CCSDS 520.0-G-3. Washington, D.C.: CCSDS, December 2010.
- [4] *Voice and Audio Communications*. Proposed Recommendation for Space Data System Standards. Forthcoming.
- [5] *Motion Imagery and Applications*. Issue 2. Report Concerning Space Data System Standards (Green Book), CCSDS 706.1-G-2. Washington, D.C.: CCSDS, May 2015.
- [6] *Mission Operations Monitor & Control Services*. Issue 3. Draft Recommendation for Space Data System Standards (Red Book), CCSDS 522.1-R-3. Washington, D.C.: CCSDS, March 2014.

- [7] *Mission Operations Common Object Model*. Issue 1. Recommendation for Space Data System Standards (Blue Book), CCSDS 521.1-B-1. Washington, D.C.: CCSDS, February 2014.
- [8] *Mission Operations Common Services*. Proposed Recommendation for Space Data System Standards. Forthcoming.
- [9] *CCSDS File Delivery Protocol (CFDP)*. Issue 4. Recommendation for Space Data System Standards (Blue Book), CCSDS 727.0-B-4. Washington, D.C.: CCSDS, January 2007.
- [10] *Space Data Link Security Protocol*. Issue 1. Recommendation for Space Data System Standards (Blue Book), CCSDS 355.0-B-1. Washington, D.C.: CCSDS, September 2015.
- [11] *Security Architecture for Space Data Systems*. Issue 1. Recommendation for Space Data System Practices (Magenta Book), CCSDS 351.0-M-1. Washington, D.C.: CCSDS, November 2012.
- [12] Mike Botts and Alexandre Robin, eds. *OGC® SensorML: Model and XML Encoding Standard*. OGC 12-000. Wayland, Massachusetts: Open Geospatial Consortium, 2014.
- [13] C. Matthew MacKenzie, et al., eds. *Reference Model for Service Oriented Architecture 1.0*. OASIS Standard, 12 October 2006. Burlington, Massachusetts: OASIS, 2006.

2 INTRODUCTION TO TELEROBOTIC OPERATIONS

2.1 OVERVIEW

Extending human presence and capability into space will require collaboration between mixed teams of human and robotic assets; robots will be tasked with the ‘dull, dirty, or dangerous’ work that is not sensible or necessary for humans to perform. Current space telerobotic operations concepts do not scale well beyond the ‘one operations team, one robot’ approach, resulting in an inability to efficiently scale operations to multi-robot teams. Including humans in mixed human-robot teams adds the additional complexity of human safety concerns, which generally further complicate the telerobotic operations concept by imposing constraints on the robotic elements, such as keep-out zones and movement speed and force limitations. Finally, the most significant barrier to international cooperation in space telerobotics is the basic inability to intercommunicate; there is no common language by which a diverse set of human and robotic collaborators can share information for the purpose of achieving a common goal.

2.2 SUMMARY OF TELEROBOTIC OPERATIONS

This subsection contains a description of the Mission Operations Services Concept and the Service Oriented Architecture upon which the concept for telerobotic operations is based. The concept layers depicted in figure 2-1 are based on the notional model of telerobotic operations described in this subsection.

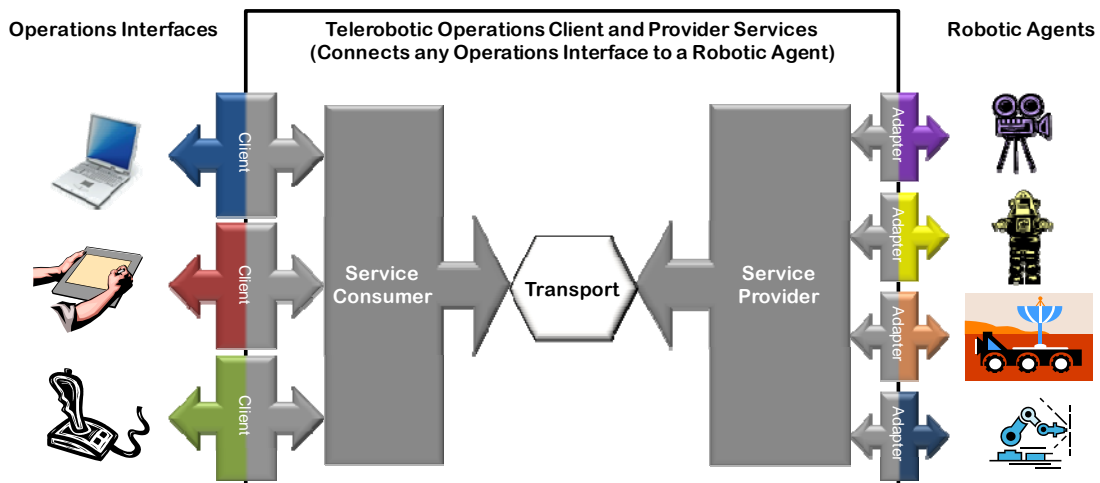


Figure 2-1: Telerobotic End-to-End Model

The list below describes some of the overall telerobotic behaviors and service dependencies described in this document. Some of the listed services are the subject of other CCSDS standards—standards upon which telerobotic operations will depend, but will not define. A complete list of services can be found in 3.2, Telerobotic Operations Services.

- a) Robotic devices (on the right in figure 2-1) publish telemetry (including, potentially, streaming audio and/or video) to one or more subscribers, which may include both human operators and other robotic devices.

- b) Human operators (on the left in figure 2-1) send operational command data privately to selected specific robotic devices.
- c) Under some conditions, human operators might publish information broadly to one or more robotic devices. Such information might include environmental data (e.g., updated maps) or contextual data governing local autonomous behavior (e.g., goals, changes in policy, revised alarm limits, etc.).
- d) Human operators may need to communicate among themselves, by text message and/or by streaming audio (see reference [4]) and/or video (see reference [5]). This communication may be either private (directed from one human operator to another) or public (directed from one human operator to a set of others, where the set of receiving operators may be self-selected—i.e., subscribers—or may be selected by the sender).
- e) Some subset of this communication may require authorization, authentication, and/or confidentiality.
- f) Some subset of this communication must be assured by acknowledgment and retransmission procedures – but some must be delivered in the order in which it was transmitted, without delay, and therefore should not be subject to acknowledgment and retransmission.
- g) Both interactive (real-time, including haptics and force-feedback) operation of proximate connected robotic devices and policy-driven (autonomous) operation of remote or temporarily disconnected devices must be accommodated by telerobotic operations. Services must be robust to a wide range of time-delayed and disrupted communications.
- h) Centralization of the monitor and control function is possible, depending upon the architecture of the communication network and the needs of the exploration system. A centralized system would integrate one or more of the left hand or right hand elements in figure 2-1 into more capable services.

2.3 OVERVIEW OF TELEROBOTIC OPERATIONS LAYERS

The telerobotic operations model utilizes a Service Oriented Architecture defined by the CCSDS. Telerobotic operations will act as a services layer just below the device-specific displays and machine/device layer at either end of the telerobotic operations as shown in figure 2-1.

The CCSDS Mission Operations (MO) layer will provide end-to-end communication and Monitoring and Control services to specific telerobotic services such as discovery, robot command, and robot telemetry, etc. The Spacecraft Monitoring and Control (SM&C) services are part of the overall Mission Operations Services that will be described in section 4, Overview of the Mission Operations Service Framework.

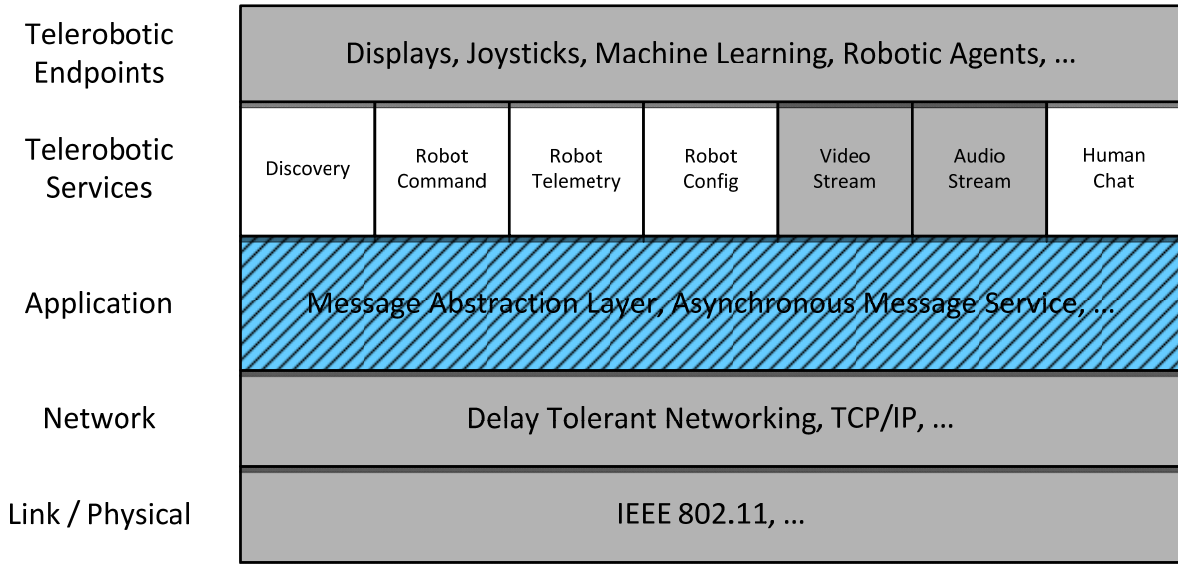


Figure 2-2: A Strawman Gap Analysis

In figure 2-2, protocols for which well-established standards exist are shown in solid gray. Protocols that are under development within CCSDS are shown with a diagonal line pattern on a blue background. Protocols shown in white are needed for telerobotics operations. The precise structure of the network stack is dependent upon the deployment or implementation; conceptual layers can be merged for performance reasons or other constraints. This will likely be the case for deployments onboard spacecraft and in a robotic system with limited computing resources.

2.4 POTENTIAL BENEFITS OF AN INTERNATIONAL STANDARD

There is a strong international desire to collaborate on defining proposed Recommended Standards for telerobotic operations in order to reduce the lifecycle costs associated with interoperability and cross-agency support in space exploration.

Spaceflight is costly across the development, flight unit production, and launch and operation phases of missions. Spaceflight is also risky to both humans and machines. Through collaboration, the international community can contribute to research that will reduce cost and risk. An even greater benefit is realized when these new technologies increase capabilities or add whole new functions that extend the possibilities of space exploration.

The savings and risk reduction obtained through the development of any component telerobotic technology is multiplied by the opportunity that interoperability offers to directly measure and compare similar technologies without a combinatorial increase in development cost. Telerobotic interoperability would allow component technologies to be tested in a rich shared environment, such as an International Space Station (ISS)-based test-bed, without the need to create new infrastructure to support each new technology.

2.5 RELATIONSHIP TO THE MISSION OPERATIONS SERVICES CONCEPT

The telerobotic operations model is, in many ways, an extension of the Mission Operations Services Concept (reference [3]); the reader who is unfamiliar with the latter subject is directed to read the overview material presented in section 4, Overview of the Mission Operations Service Framework.

Each telerobotic operations service provides a set of well-defined capabilities through a standardized service contract (i.e., the interface, specified in a proposed Recommended Standard). The telerobotic operations service contracts will be specified in an implementation and communication technology-agnostic manner. For example, a Motion Control service would provide the capability to *move* a robotic asset to a particular position (absolute or relative), without making any assumption about what programming language is used by the service provider or service consumer and without making any assumptions about what communication technology would be adopted in a particular deployment scenario to establish the link between the service consumer and service provider.

Each capability is specified in form of an abstract operation, which itself is defined by a set of exchanged messages between the service consumer and service provider, following a prescribed interaction pattern. The messages exchanged between the service consumer and the service provider would contain both *data* (message body) and *metadata* (message header). The data part of all messages exchanged between the service consumer and provider for all capabilities of a service together compose the *data or information model* of a service.

The metadata part of the messages (the header) captures usually the non-functional aspects of interactions between the service consumer and the service provider, such as the aspects related to these and others:

- **Addressing:** How to reach the service consumer and service provider;
- **Service Taxonomy:** The domain, and the name and version of the service;
- **Transaction Management:** How to correlate messages in asynchronous interaction patterns;
- **Security;**
- **Quality of Service:** Also known as Service Level Agreements.

In a concrete deployment scenario, the service consumer and service provider must implement the abstract (i.e., technology independent) service contract, in a concrete technology (e.g., a programming language such as Java or C, ADA, C++ or .NET). In the selected programming language, the abstract service contract is typically realized through a language-specific Application Programmers' Interface (API), an interface definition in the form of operations or methods and data types.

In addition, the service consumer and service provider will use a concrete communication technology for the exchange of messages at run-time that is suitable for that particular deployment scenario (e.g., pure TCP/IP, DDS, HTTP over TCP/IP, SOAP over HTTP over TCP/IP, CCSDS Space Packet binding over AMS over DTN, etc.).

3 DEFINITION OF TELEROBOTIC OPERATIONS

3.1 INTRODUCTION

This section describes the core of the information interface, processing, and access methods to be defined in a proposed Recommended Standard. These interface elements are described at the informational level, leaving the formal specification of the corresponding service interfaces—in terms of operations signatures, interaction patterns, message and data formats, as well as error and exception handling—for a proposed Recommended Standard.

3.2 TELEROBOTIC OPERATIONS SERVICES

3.2.1 OVERVIEW

In order to specify generic, interoperable telerobotic operations services, the core of a proposed Recommended Standard will be structured into the main elements of a Service Oriented Architecture: services, operations, messages and data types:

- Services are to be expressed as a collection of operations; each operation provides a particular telerobotic capability that is relevant for interoperable collaboration.
- Operations are to be expressed as a specific well-defined pattern of exchanged messages between the two or more parties involved (i.e., the service provider and service consumers), in order to achieve the corresponding collaboration capability.
- Messages are to be specified as a collection of data types exchanged for each interaction step.
- Data types are to be provided in the formal specification for the information exchanged through messages.

Telerobotic operations services are defined, in part, by the information they carry between members participating in collaborative telerobotic operations. That information is roughly divided into two categories: information conveyed as part of a commanding operation, and information conveyed as part of a monitoring operation. Services are further divided into two broad categories: those that are specific to telerobotics, and those that have a broader applicability. Those with more broad applicability are likely to be the subject of other standards definitions.

3.2.2 TELEROBOTIC OPERATIONS SCENARIOS

3.2.2.1 Introduction

This subsection highlights some elements of human and robotic exploration activities and how they might utilize the telerobotic operations services described in this document. The scenario descriptions include both functional elements (for instance, keep-out zones, which would utilize the Frame Store Service to monitor collaborator locations and warn of

incursions) and performance elements (for instance, data rates and communication latency). Scenarios that reference advanced capabilities, such as optical communications, should also include a roadmap for qualifying the technology for mission use. In developing these and any future operations scenarios, an attempt should be made to ensure that each of the services described in this document is referenced in at least one operations scenario.

The operations scenarios listed in this subsection are not exhaustive; more potential operations scenarios might be described in support of future exploration plans and architectures developed by member agencies. The reader is encouraged to consult agency and international roadmaps, such as the Global Exploration Roadmap (reference [2]), to establish an alignment between the scenarios described in this subsection and the larger context described in the exploration roadmaps.

Highly detailed scenario descriptions are outside the scope of this document, but the reader is encouraged to consider the aspects of any scenario that might impact the use or suitability of any of the telerobotic services described in this document.

The following operations themes may be considered:

- Inter- and Intra-Agency Collaboration:
 - Testing:
 - Field or Laboratory;
 - Simulation Services;
 - Emergency Support;
 - Nominal Operations Support;
 - Research and Technology Development;
 - General Service Level Agreements:
 - Time and Resource Sharing;
 - Data Sharing and Embargo Policies;
 - Communications Infrastructure;
- Human / Robot Collaboration:
 - Human Safety;
 - Operations Efficiency;
 - Number and Distribution of Operators and Agents;
 - Venue: IVA, EVA, Surface;
- Suitability:

- When is it **not** appropriate to use telerobotics operations, possibly because of degraded network capability, etc.;
- Time Delay Regimes:
 - Negligible—No appreciable time delay;
 - 850 ms—Force-reflection performance boundary;
 - 2 s—Earth-Moon time delay;
 - 50 s—Earth-NEO time delay;
 - 20 m—Earth-Mars time delay;
- Mission Destinations:
 - Earth orbital;
 - Cislunar;
 - Lunar surface;
 - Mars orbital;
 - Mars surface;
- 5-, 10-, 25-year development and exploration schedules.

3.2.2.2 Motion Imagery Scenario

Video and motion imagery systems, particularly when designed for external spacecraft applications are, in essence, simple robotic systems. Typical systems would likely include a camera with a zoom lens, mounted onto a platform that can be panned (moved left to right) or tilted (pointed up or down). The camera itself will likely have several functions that need to be controlled and monitored, such as iris or exposure, white balance, and other modes such as frame rate and resolution. The lens would need to be focused and the focal distance changed (zooming in or out). The mount, or pan-tilt unit, would also need to be controlled or automated. Metadata from the camera system would be very important, enabling viewers downstream to verify the camera's position in three-dimensional space, or providing indicators for where the camera is pointed on an x- and y-axis. Operating temperature and other similar data would also be critical to ensure proper operation.

To date, most video and motion imagery systems flown on spacecraft have utilized unique or proprietary interface controls, software, and/or hardware to manipulate the camera system. This proliferation of control approaches leads to inefficiencies, increases costs, and reduces redundancy and interoperability. Development of common services and protocols that could be reused for video and motion imagery systems would be highly desirable as future missions are likely to involve multiple space agencies, private enterprises, or combinations of both.

Among the challenges of incorporating video and other motion imagery data in telerobotic operations is that these data sources consume a large portion of the bandwidth available in space communications links. Advances in motion imagery compression will improve the situation somewhat, but those advances will likely be offset by the increased use of virtual telepresence systems for operations that include immersive technologies such as three-dimensional video.

The subject of Virtual Telepresence as part of the a proposed Recommended Standard for telerobotic operations is an open question, discussed in 3.4.5, Virtual Telepresence.

3.2.2.3 Dissimilar, Redundant Robots

In future robotics exploration scenarios, but also inside and outside orbital stations in LEO, near Moon, or Mars, several robots may be working together. In the foreseeable future, there will always be just a few people around, with a very high workload. Robots that are able to perform tasks designed for humans will be very useful to take away some of this workload, but will also have a role in performing dangerous and/or repetitive tasks. Robots are getting more and more autonomous, but still some level of monitoring by humans will be foreseen and for some tasks also teleoperation will be useful.

Especially when going beyond LEO, any failure in a robotic system would be a major issue, as onboard repair and replacement would be impossible, or at least very costly. The reliability on a mission level will be significantly increased if the robots are of dissimilar design, to minimize the risk of common-cause failures due to unforeseen design flaws or wear.

Bringing together robots from several space agencies will increase the number of robots available for work in space as well as provide dissimilar redundancy for single robot operations. An example scenario could be the use of robots like NASA's Robonaut, ESA's Eurobot, and Russia's SAR-401 on LEO or cislunar space stations. The same dissimilar redundancy rationale could, although to a lesser extent, also be applied for the reliability of the robots' monitoring and control stations, which, in case of master/slave control systems, could also be of significant complexity.

To enable the above scenarios, workstations and robots would need to follow standardized interfaces and operations concepts, and therefore be interoperable. Robots could be each other's backup, possibly even a hot backup. A proposed Recommended Standard for telerobotic operations is expected to make a contribution to such a future goal.

3.2.2.4 Shared Autonomous Control

In the future, planetary exploration architectures may involve a surface-based robotic device that is controlled from an orbiting spacecraft. Tasks that cannot be executed autonomously would be performed by direct teleoperation in real-time. Tasks that can be performed with a higher autonomy would be invoked and supervised. Hence, this scenario concerns a mixed form of direct real-time teleoperation with high-level commands to invoke autonomous task execution.

Real-time teleoperation

The interaction between operator and robotic asset is direct and continuous, with immediate execution of operator inputs. The main characteristics are:

- real-time commands;
- real-time feedback from the robotic assets (telemetry, video); and
- stepwise-execution with a high frequency.

In this context real-time means a latency less than 100 ms. In case of interruption through signal loss or increasing latency the operation will no longer be feasible or executable. The typical example is a repair or installation procedure.

Semi-autonomous operations

The interaction between the operator and the robot takes place in a less direct way. The operator initiates procedures and planning, and the execution is done autonomously. The main characteristics are:

- high-level commands to initiate autonomous executions;
- supervision by the operator; and
- task-wise execution.

Delayed or interrupted communication has little or no consequences on the task execution. Examples include:

- perform measurements at different locations and transmit results after completion;
- collect items from distributed locations.

Combination of real-time teleoperation and semi-autonomous operations

Certain tasks will require a combination of real-time teleoperation and semi-autonomous operations, for example, the inspection and maintenance of distributed locations on the planetary surface. In this case the robot would autonomously navigate and respond upon reaching the location destination. The local maintenance task at the destination would be done via direct teleoperation from the operator in the spacecraft.

ESA's Haptics-2 and Interact Experiments

In September 2015, Haptic-2 tested the real-time link from an operator on the ISS to a simple robotic asset on Earth, and Interact tested real-time teleoperation and semi-autonomous operations of the operator in space on a four-wheeled rover complete with robotic manipulators on Earth. The input control hardware that the user used to remotely control the robotic asset in Haptics-2 and Interact consisted of a 1-DOF high-resolution force-feedback joystick complete with tablet.

The link providing low-latency was a line-of-sight S-band connection with 256 kb/s data rate and a 100 ms delay. The high bandwidth link was the ISS KU-forward link with ≈ 1 Mb/s bandwidth and a higher latency of approximately 2 seconds. Independent from the physical link, DDS was chosen as the real-time data-centric communication middleware.

These experiments consisted of a set of semi-autonomous and directly controlled tasks that utilized real-time bilateral control, which is robust to temporary loss of network connection. These experiments can be used as a test platform and/or reference for future CCSDS implementations.

3.2.2.5 Control of Multiple Agents Simultaneously

As indicated in 3.2.2.3, scenarios are anticipated where robots and people will work together. In principle, one person can be leading several robots. The robots are autonomous to a certain level. They do not continuously need guidance, monitoring, and control, but they need it depending on the task they are executing at any moment in time. The consequence of such a scenario for the system architecture is that it would be ideal for a single workstation to be able to control multiple robots simultaneously, be it from Earth, from a zero-gravity orbiter or space station, or from a surface base. This would be enabled by standardization of interfaces to achieve interoperability of robots.

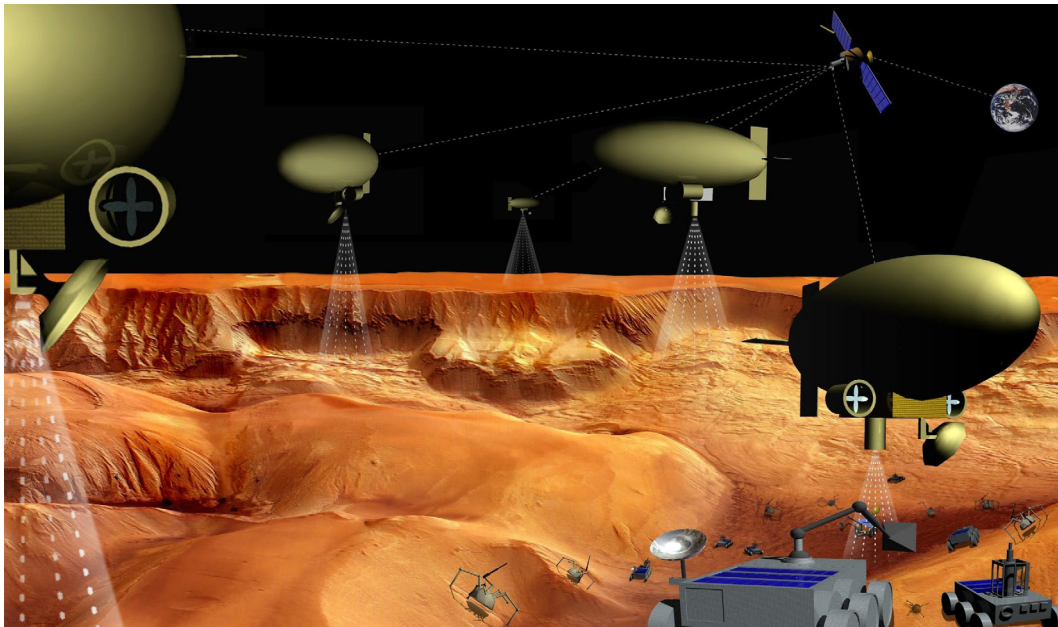


Figure 3-1: Artist's Concept of Orbiter, Blimps, Rovers, and Robots Working Together¹

¹ Image Courtesy NASA/JPL-Caltech.

The control of multiple agents simultaneously is particularly important in an exploration concept known as robot swarms. Swarm robotics is a relatively new technique that applies systems of multiple identical or nearly identical robot agents to the solution of a complex problem. In field robotics, such as the exploration of the Martian surface, a system of similar robots can cover a larger percentage of unknown terrain in a shorter duration, increasing the amount of time spent exploring any single target, and improving the odds for making the chance discovery. In orbital robotics, swarm robotics often appears in the literature as networks of formation flyers. As scientific platforms, formation flyers can be used to increase the effective baseline of an observation platform, such as in synthetic-aperture radars or high-resolution remote sensing platforms. Formation-flying technologies are also being applied to fractionated spacecraft, where the functions performed by a single monolithic spacecraft are partitioned to a swarm of independent spacecraft acting as functional modules of a system. Among the benefits of swarm techniques is the reduction in risk from the loss of any single agent, allowing the swarm system to be used to accept more risk and operate in more challenging environments.

In many cases, individual elements of a swarm are tightly linked to other elements through common spacecraft design, relative positioning requirements, and data system interfaces. Currently, swarm systems are generally produced by a single provider and utilize a custom flight/ground interface and control methodology. Applying standards in telerobotic operations services can open up the swarm system to multiple agent providers, increase cross-agency support, and enable the control of multiple swarm systems through standard ground control station technology.

3.2.2.6 Inter-Agency, Inter-Center Field Testing

International space agencies are planning to expand human and robotic exploration beyond LEO to destinations including the moon, near-Earth asteroids, and Mars and its moons. To prepare to explore these destinations, agencies are first conducting analog missions here on Earth. Analog missions are remote field tests in locations that are identified based on their physical similarities to the extreme space environments of a target mission. Working together, engineers and scientists from government agencies, academia, and industry gather requirements, develop technologies, and test robotic equipment, vehicles, habitats, communications, and power generation and storage. Field tests are designed to evaluate mobility, infrastructure, and effectiveness in the harsh environments.



Figure 3-2: ESA Rio Tinto Field Test in Andalusia, Spain

Analog missions provide an excellent opportunity to perform interoperability testing of assets from multiple agencies, as well as a representative environment for testing a proposed Recommended Standard for telerobotic operations. The harsh environment and remote location of the field test site often present real-world challenges to the coordination and communication required to meet mission objectives. Heat, dust, and terrain present localized communication challenges, often resulting in the kinds of disruption that complicates space links; the remote location often involves long backhaul connections to the Internet and related communication delays. Communication delay is often injected as a test condition when delays of tens of seconds are required to simulate distances to near-Earth objects and beyond.

To control costs, analog missions incorporate realistically designed elements of the space exploration system only where such fidelity serves a specific analytical need. In some cases, commercial-off-the-shelf components are used as functional equivalents of their space-qualified siblings. For instance, analog missions often use commercial GPS units for localization in place of more expensive space-qualified hardware (in this example, GPS technology is currently Earth-centric). A similar approach can be used in testing a proposed Recommended Standard for telerobotic operations in analog missions by simplifying the communications stack and operating the message and service standard over the terrestrial field test network with commercial technology such as RTI DDS over UDP.

Where terrestrial test network infrastructure currently exists to support inter-agency communication, a proposed Recommended Standard can be utilized over the commercial Internet between agency-development laboratories. Use of existing infrastructure will reduce the costs associated with the independent prototyping and test activities required of CCSDS standards-development groups.

In addition to opportunities to test a proposed Recommended Standard for telerobotic operations, field testing offers favorable circumstances for testing cross-agency support, including emerging standardization areas such as scheduling and mission planning. Field testing infrastructure can also be modified to support collaborative use of agency testbed infrastructure, such as air-bearing floors, reactive robot bases, macro/micro robotic system, Stewart platforms, etc. The same networks that transport data between field test sites and home agencies can also be used to connect partner agencies' testing facilities.

Examples of analog field testing can be found at the NASA Desert Research and Technology Studies Web site and in literature about the ESA Eurobot test at Rio Tinto.

3.2.3 NETWORK ARCHITECTURE

Collaborative telerobotic operations necessarily involve multiple operators and multiple robots operating on a single connected network. While the architecture of the network is not necessarily the subject of this document, certain capabilities of the network can greatly simplify the telerobotic operations that the network supports. A proposed Recommended Standard for telerobotic operations will generally reference the CCSDS MO SM&C Operations Domain to help define the mission network operations domains.

Some of the capabilities required include the ability to enable the monitoring of a single Agent's telemetry by multiple Operator stations (and other Agents); the network should support a one-to-many distribution model, such as multicast, in addition to point-to-point transmission. Scenarios in which this type of distribution model would be beneficial include a lunar surface robot being monitored by an adjacent suited astronaut, in addition to an IVA support astronaut in a nearby lunar outpost, and Earth-based operators.

3.3 TELEROBOTIC OPERATIONS SERVICES

3.3.1 INTRODUCTION

This subsection defines and categorizes the classes of information exchanged between participants. A proposed Recommended Standard for telerobotic operations should define common capabilities that greatly increase the ability of Agents to collaborate on common telerobotic tasks. In cases where a service is an extension or subclass of an existing CCSDS service, the CCSDS service is called out in the referenced section.

The following services are defined in this subsection:

- The Manipulation and Limbed Mobility Service controls the motion of manipulators such as robot arms, legs, booms, sample acquisition devices, and cameras.
- The Sequencer Service enables time-delayed teleoperation of robotic agents through a synchronous command queue.
- The Frame Store Service provides location awareness between robots.

- The Asynchronous File Transfer Service provides a robust file delivery mechanism between Agents.
- The Access Control Service authenticates Agents, authorizes their participation in the Agent network, and ensures the security and reliability of data in transit.
- The Transfer of Control Service mediates requests by Operators to control Agents by ensuring that requests to transfer control between Agents are handled according to an established policy.
- The Task Decomposition Service provides traceability from high-level tasks, down through their decomposition into lower-level components, such as operations and actions, to the telemetry produced by those actions.
- The Imaging Service provides capabilities for capturing images from Agent-mounted cameras.
- The Video Service provides moving images to the Operator as a natural way of representing the dynamic state of the environment in which the Agent is operating.
- The Command Service provides a signal that, when sent to an Agent, causes the Agent to perform a function not otherwise covered by the services described in this document, and for which some coordination is required between Agents and/or Operators.
- The Direct Command Service provides a signal that, when sent to an Agent, causes the Agent to perform a function not otherwise covered by the services described in this document, and for which no coordination is required between Agents and/or Operators.
- The Data Product Service provides type-specific access to operationally relevant data collected by Agents.
- The Discovery Service provides a method by which to find Agents and their capabilities as they join and leave the network.
- The Administrative Service provides system-level managerial and organizational functions for test and control.
- The Location Service establishes the relative or absolute position and velocity of Agents during surface or free-flight operations.
- The Wheeled Mobility Service controls the motion of navigable rovers over a surface.
- The Configuration Service provides information for describing the existence and state of various components that affect Agent operations.

Agents will generally be required to support a minimum set of services in order to qualify as adhering to a proposed Recommended Standard. Different sets of capabilities may be appropriate for different classes of Agents: navigable rovers might be required to provide the Location Service, for instance. Required elements will tend to support the ‘mission statement’ of a proposed Recommended Standard: safe, supervisory, and collaborative telerobotics over time-delayed and disruption-prone networks.

3.3.2 MANIPULATION AND LIMBED MOBILITY SERVICE

The Manipulation and Limbed Mobility Service controls the motion of manipulators such as robot arms and legs, booms, sample acquisition devices, cameras, and other devices that require articulated control. The Manipulation and Limbed Mobility Service uses the Command Service to effect control over a wide variety of types of manipulators.

Agent manipulation is normally accomplished via discrete commands within the context of having established normal parameters of the manipulation action, such as controller gains, force thresholds, etc. The Command Service provides access to standard mechanisms for executing actions, and for selecting from among a finite set of well-understood performance specification parameters.

The Manipulation and Limbed Mobility Service can be further broken down into hierarchical parts, each addressing different levels of complexity in manipulation planning, such as task, operation, and action.

Further development of this hierarchical approach is left for subsequent issues of this document, and for further development of use cases in a proposed Recommended Standard. In general, however, the Manipulation and Limbed Mobility Service should allow for an abstract approach that allows, for example, legged robots whose limbs can provide both mobility and manipulation (e.g., NASA's ATHLETE).

3.3.3 SEQUENCER SERVICE

The Sequencer Service is also referred to as the Onboard Queue Management Service. It enables time-delayed teleoperation of robotic agents through a synchronous command queue and provides the capabilities for managing that queue remotely by adding, removing, or modifying commands in the queue.

The Sequencer Service uses the MO Configuration Service (reference [8]) to manage a queue of commands to be executed by a designated agent, including Queue Control and Queue Report functions. The Queue Control operation set includes methods for managing the Sequencer Service's queue of commands, including the insertion and deletion of commands from the queue. The Queue Report operation set includes methods for providing status messages that reflect the Sequencer Service's current state.

As the capabilities related to execution of an onboard time-tagged command queue and those pertaining to managing such a queue remotely are most likely very similar to those of onboard queue management of spacecraft, this service may belong to a category of 'non-telerobotics-specific' services.

3.3.4 FRAME STORE SERVICE

The Frame Store Service provides location awareness between robots by enabling inter-Agent exchange of kinematic and coordinate frame information. The service provides a data store for relevant objects and a processing capability for computing spatial relationships between objects. The service generally provides location awareness between Agents, and is generically implemented as a classic tree of coordinate frames and tree-walking routines for calculating coordinate transforms. The service data stores are generally updated from Agent telemetry.

The Frame Store Service uses the MO Configuration Service (reference [8]) to manage an Agent's kinematic and coordinate frame information, including the definition of the location of an Agent's linkages and joints, symbolic naming of coordinate frames, and linkages in coordinate frame trees.

The Frame Store Service also provides a compute service for calculating relative offsets between coordinate frames within an Agent as well as for calculating relative offsets between coordinate frames between Agents.

3.3.5 ASYNCHRONOUS FILE TRANSFER SERVICE

The Asynchronous File Transfer Service provides a robust file delivery mechanism between Agents. The service additionally supports file-based operations in support of tactical and strategic telerobotic operations, including file abstract processing, file directory service and metadata processing, and file-related quality of service specifications for compression and latency.

The Asynchronous File Transfer Service makes use of the CCSDS File Delivery Protocol (reference [9]) for basic file transfer operations.

3.3.6 ACCESS CONTROL SERVICE

The Access Control Service authenticates Agents, authorizes their participation in the Agent network, and ensures the security and reliability of data in transit. The service ensures that Operators have the proper security credentials to access the Agents under control and that communications between Agents are free from unauthorized modification in transit. The Access Control Service is a fundamental service for telerobotic operations and is a precondition for nearly all other services described in this document.

The Access Control Service will depend upon other CCSDS information security protocols, including:

- Space Data Link Security Protocol (reference [10]);
- Security Architecture for Space Data Systems (reference [11]).

3.3.7 TRANSFER OF CONTROL SERVICE

The Transfer of Control Service mediates requests by Operators to control Agents by ensuring that requests to transfer control between Agents are handled according to an established policy. Transfer of control policies will likely differ according the relative locations of Agents and Operators (local versus remote) and Agent operating mode (test versus operations).

The Transfer of Control Service uses the MO Configuration Service (reference [8]) to establish the policies under which control may be transferred as well as to indicate the present state of control, i.e., which Agents are controlled by which Operators.

It should be noted that while security policy is out of scope of this service, the Transfer of Control Service shall include a policy that controls the authorization of transfer of control.

3.3.8 TASK DECOMPOSITION SERVICE

The Task Decomposition Service provides traceability from high-level tasks, down through their decomposition into lower-level components, such as operations and actions, to the telemetry produced by those actions. The Task Decomposition Service helps structure the relationship between action and reaction in a complex telerobotic operations system. Much like the Frame Store Service, the Task Decomposition Service can be thought of as a tree-oriented structure with high-level tasks as the root nodes and low-level commands and telemetry as the leaf nodes. The Task Decomposition Service contains registries of specific task decompositions, which can then be associated with specific instances of the execution of those tasks. By referring to the registry, systems can reassemble the structure of an executed task from the telemetry provided by the Agent for the purpose of presenting the observed behavior of the Agent in a meaningful context. The Task Decomposition Service will be especially useful when reconstructing behavior taking place over varying time-delay, which results in a complex series of overlapping command and telemetry messages.

3.3.9 IMAGING SERVICE

The Imaging Service is used to capture images from Agent-mounted cameras, and provides information about Image-related products, such as metadata, data, resolution, planes, etc.

The Imaging Service relies upon the Command Service to command the acquisition of images, and determine the current state of Agent-mounted imaging devices. In addition to the image data itself, the Imaging Service provides camera and image metadata, such as field of view, image dimensionality, and other data describing the acquired image.

3.3.10 VIDEO SERVICE

The Video Service provides moving images to the Operator as a natural way of representing the dynamic state of the environment in which the Agent is operating. Telerobotic operations naturally involve the motion of robotic agents and manipulation of their environment, and moving images, or video, is a natural way to represent the state of the environment to an operator. Issues of time delay and communication disruption make the presentation of motion imagery (reference [5]) challenging in terms of filtering out irrelevant data (old, poor quality, etc.) from useful imagery.

The Video Service also provides a method for transmitting non-real-time video and cataloging video for eventual transmission as part of scientific or diagnostic activities. The Video Service shares some characteristics of a still imagery service in that the imagery is provided by a specific device or sensor that is often mounted on an articulating platform that can be controlled as part of the telerobotic operations system. Video and still imagery data share quality-of-service attributes that affect the value of the imagery for tactical, strategic, and scientific uses (compression, resolution, drop-out, etc.).

The Video Service makes use of the Manipulation Service to control articulating platforms.

The Video Service makes use of the Imaging Service in areas where functionality overlaps, such as in device selection, resolution, optical zoom, etc.

The Video Service uses the MO Configuration Service (reference [8]) to enable the following operations:

- **Source Configuration:** Resolution, refresh rate, etc.;
- **Source Selection:** Enabling and disabling of cameras as active data sources;
- **Stream Control:** Quality of service, minimum and maximum allowable compression, minimum and maximum data rates.

3.3.11 COMMAND SERVICE

The Command Service provides a signal that, when sent to an Agent, causes the Agent to perform a function not otherwise covered by the services described in this document, and for which *some* coordination is required between Agents and/or Operators. The Command Service is analogous to the MO Action Service (reference [7]), but with additional attributes to support the highly interactive nature of telerobotic operations. The Command Service utilizes both the MO Action Service (reference [7]) and the MO Configuration Service (reference [8]) to provide its functions.

The Commands function provides a list of Commands that are supported by the receiving Agent, and extends the MO Action Service (reference [7]) by adding attributes for:

- **Priority:** Insertion point within the Agent’s command queue. Choices are Low, Medium, High, and Asynchronous. The default is Asynchronous, unless the Agent is using the Sequencer Service.
- **Name:** Name of the Command being sent; must be one of the Commands included in the receiving Agent’s Commands Configuration Message.
- **Unique ID:** Unique identifier for this instance of a Command. One possible unique identifier would be a concatenation of the sending Agent name and the timestamp of the sending event.
- **Source:** Identifier for the source of the command. The value for this attribute is arbitrarily chosen to aid in tracing the execution of commands. Source identifiers might include the subsystem or process that generated the command within the sending Agent.
- **Target Subsystem:** Unique identifier for the subsystem within the receiving Agent that is to receive the Command.
- **Arguments:** Name, data type, and value of the arguments for this particular command.

3.3.12 DIRECT COMMAND SERVICE

The Direct Command Service provides a signal that, when sent to an Agent, causes the Agent to perform a function not otherwise covered by the services described in this document, and for which *no* coordination is required between Agents and/or Operators. For instance, operation of a unique Agent-mounted instrument, such as a laser sinter, would not be included in a proposed Recommended Standard for telerobotic operations. It would describe services beneficial to safely operating collaborative human-robot teams in space, but would not address potentially unique Agent characteristics. The Direct Command Service provides a ‘pass through’ mechanism whereby unique Agent capabilities can be addressed within the context of a telerobotic operations service standard without needing to standardize operations of unique functions.

The Direct Command Service inherits from the Command Service many required attributes, such as Unique ID, Source, and Target, but many of the remaining attributes become optional, such as Priority, and the values used for the parameters become unchecked, such as the Command Name.

3.3.13 DATA PRODUCT SERVICE

The Data Product Service provides type-specific access to operationally relevant data collected by Agents. The service is extensible to cover current and future data product types. Some of the current operationally relevant data products are:

- Point Clouds: Useful for providing depth data for object recognition for manipulation and terrain maps for navigation route planning. Point Cloud formatted data is also used in the collection of scientifically useful information. Point Cloud data can be generated by a variety of devices, including LIDAR, light-field cameras, and stereo cameras.
- Digital Elevation Maps: A digital model or 3D representation of a surface, typically the Earth, Mars, or the Moon: common surfaces that can be navigated by wheeled rovers.
- Traversability Maps: A digital representation of hazards or other unsuitable areas on a surface, indicating areas where a wheeled or limbed robot may wish to avoid, for platform-dependent reasons. For instance, soft sand may prove to be a hazard to rovers with small wheels; steep slopes may be hazardous to a limbed robot.

This Data Product Service is closely related to MO Mission Data Product Distribution Services, which were being defined when this document was issued.

3.3.14 DISCOVERY SERVICE

The Discovery Service provides a method by which to find Agents and their capabilities as they join and leave the network. The Discovery Service maintains a local store containing information on all Agents who are part of the current network environment. Agents may join and leave the network at will, and they broadcast their presence in support of a discovery mechanism. Individual capabilities may also join and leave the network at will.

For instance, a specific camera may become unresponsive and be marked either as unusable or as no longer part of the capabilities of the Agent that once hosted the camera. Physical addition and removal of devices is a more common occurrence in environments with a mix of human and robotic Agents.

The discovery process also applies to the configuration of the communication network itself. For instance, certain types of message traffic may be partitioned from other types of message traffic to aid in network management. The Discovery Service also supports methods for finding these partitions as they are created and destroyed.

The Discovery Service provides a basic aliveness service that periodically ‘pings’ Agents to determine their availability.

The Discovery Service is similar in functionality to the Directory Service, as described in Mission Operations Common Services (reference [8]), and may be replaced in a future issue of this document.

3.3.15 ADMINISTRATIVE SERVICE

The Administrative Service provides system-level managerial and organizational functions for test and control, such as Echo, No-operation, or Shutdown. The Administrative Service may utilize the MO Action Service (reference [7]) to execute its functions and the MO Configuration Service (reference [8]) to manage the behavior of its functions.

Potential Administrative Service functions include:

- Echo function to retransmit the received payload to the originating Agent;
- Shutdown function to terminate all connections to the receiving Agent and removes the Agent from the collaborative network; and
- Message Rate configuration function to manage Agent message publishing rate for streaming telemetry.

This Administrative Service may be deleted and its functions transferred elsewhere. For now, the Administrative Service can be considered to be a ‘catch-all’ service for items with no clear home in one of the other telerobotic operations services.

3.3.16 LOCATION SERVICE

The Location Service establishes the relative or absolute position and velocity of Agents during surface or free-flight operations. The Location Service uses the Configuration Service to maintain a database of symbolic names for Agent positions, as well as to periodically report on the location and velocity of Agents.

Potential Location Service functions include:

- Location Name function defines symbolic references to specific locations and resolves those references when requested;
- Position function reports on the position of an Agent;
- Velocity function reports on the velocity of an Agent; and
- Path function stores and reports on previous or proposed Agent Locations and is suitable for storing Agent paths.

The Location Service has some utility in managing keep-out zones for robots and humans working in close proximity. Care needs to be taken to distinguish between the Location and Frame Store Services, and between services and data models in general.

3.3.17 WHEELED MOBILITY SERVICE

The Mobility Service controls the motion of navigable rovers over a surface or through space (for free-flyers). The service provides an interface to the greatest common mobility factor for the broadest set of navigable Agents, including wheeled rovers with a variety of mobility and steering mechanism, and legged robots. The Wheeled Mobility Service provides both Move and Stop functions through its use of the Command Service.

Potential Wheeled Mobility Service functions include:

- Move function allowing for one of several different types of moves, including simple straight-line moves to complex moves with rotations and translations (moves can also be associated with tolerances when addressing Agents that are capable of assessing their own performance); and
- Stop function stops all motion of the Agent, including Joints, Wheels, etc.

3.3.18 CONFIGURATION SERVICE

The Configuration Service provides information for describing the existence and state of various components that affect Agent operations, such as number of manipulators, joints per manipulator, joint limits, etc. In many ways, telerobots are analogous to highly complex spacecraft. As such, it makes sense to consider how the Mission Operations Service Concept's Service Configuration specification might support the definition of a telerobotic operations configuration service. It is recognized that the use of the term 'Configuration Service' to refer to both the Mission Operations and telerobotic operations concepts is confusing, but the use of an alternate term in the current context would weaken the connection to the robotic use of the term 'configuration' to refer the precise physical assemblage as well as the set of robot joint angles used to reach a location in Cartesian space. To remove the ambiguity, 'MO' is prepended to terms that refer to the Mission Operations Service Concept's Service Configuration specification.

This subsection lists specific configuration elements that are of interest during telerobotic operations. These elements may also be suitable for inclusion in a Directory Service; robot configuration information is known a priori but changes dynamically during operation (losing a camera, for instance).

The Configuration Service uses both the MO Action Service (reference [7]) and the MO Configuration Service (reference [8]) to provide its functions:

- Joint Definition function provides information on the name of joints, frame references, and type of motion (rotary, linear, etc.);
- Joint Configuration function defines the kinematic chain of joints that make up a manipulator;
- Joint State function provides status on an individual joint, such as Enabled, Disabled, Stopped, Stuck, Failed, Overcurrent, Error, etc.;
- Joint Sample function provides time-varying data per joint, such as Angle, Velocity, Acceleration, Current, Torque, Temperature, and various Extended State data, etc.; and
- Dynamic Parameters function provides information on force control modes, controller modes, gains, free-floating base, etc.

3.4 OPEN QUESTIONS REGARDING TELEROBOTIC OPERATIONS

3.4.1 INTRODUCTION

This subsection lists, in no particular order, some open questions regarding telerobotic operations. It is foreseen that, as additional areas of interest arise, they can be dealt with in subsequent issues of this document.

3.4.2 LOW-LATENCY OPERATIONS

Communications latency is present in nearly all telerobotic operations. Supervisory telerobotics requires relatively low latency for safety and efficiency, but some telerobotic operations require ultra-low-latency or near-real-time communication in which latency can be measured in the low milliseconds range, e.g.,

- data streaming for direct motion control functions;
- monitoring time critical data (especially for bilateral control);
- synchronization service, especially between haptic, audio, and video streams.

All concerns relating to the use of audio and video in telerobotic operations should be discussed with the relevant CCSDS Working Groups.

3.4.3 VOICE AND AUDIO SERVICE FOR TELEROBOTIC OPERATIONS

As used in this document, voice service is considered to be real-time or near-real-time voice communications, while audio service is considered for general sounds and delayed audio and voice transmission. Voice service is generally considered to be a streaming data service, while Audio service can be streamed or file-based.

As of this issue of the document, the relationship between telerobotic operations and voice and audio services is not completely characterized. Additional testing is needed to assess the suitability of existing voice and audio standards to support the needs of telerobotic operations.

To resolve these unknowns, it will be necessary to:

- identify requirements for voice and audio streaming in support of telerobotic operations beyond those provided by Voice and Audio Communications (reference [4]) and subsequent documents;
- determine if the Voice and Audio Communications (reference [4]) and subsequent documents cover all the uses of voice communication in telerobotic operations, including flight/ground, flight/flight, etc.;
- determine if telerobotic operations needs an audio streaming service to support operator voice loop or operator-agent voice command;

- determine the role of voice or audio in coordination of operations between multiple human operators;
- determine the role of near-real-time audio in supporting telepresence, for instance, the sound of a drill in use to augment video and telemetry feedback for drill performance.

3.4.4 OTHER RELEVANT CCSDS STANDARDS

There are a number of CCSDS concepts represented in CCSDS Informational Reports that, while addressing the needs of telerobotic operations, are not currently mature enough to be included in this baseline concept document. Once the following concepts are developed in more detail, their inclusion may be revisited in future versions of a proposed Recommended Standard for telerobotic operations:

- Missions Operations Services Concepts:
 - Automation Service;
 - Scheduling Service;
 - Time Service;
 - File Management Service.

3.4.5 VIRTUAL TELEPRESENCE

Virtual telepresence technologies (including virtual and augmented reality) are being shown to improve accuracy in science operations as well as enable remote collaboration between scientists. Including support for these technologies will ‘future proof’ a proposed Recommended Standard for telerobotic operations.

3.4.6 SERVICE SUITABILITY SERVICE

It is not yet known if there is need of a service that would determine if the operations environment is suitable for executing certain telerobotic operations. This is a secondary consideration, but it can be an important consideration in determining whether or not to perform an action. For instance, NASA does not command its Mars rovers during solar conjunction because of the low likelihood that a command load will be received correctly. Other telerobotic operations may be reserved for situations with better communications coverage or data rates, for instance.

4 OVERVIEW OF THE MISSION OPERATIONS SERVICE FRAMEWORK

4.1 MISSION OPERATIONS SERVICE INTRODUCTION

The following descriptions of the Mission Operations Services Framework are derived from CCSDS Mission Operations Services references. Key elements were extracted from over 500 pages of reference material and summarized here in order to:

- summarize key MO concepts for the telerobotic operations practitioner who may not be familiar with them;
- make navigation of the MO document easier by incorporating these key MO concepts into the body of this informational report.

4.2 MISSION OPERATIONS SERVICES CONCEPT

This CCSDS Mission Operations Services Concept Green Book (reference [3]) is a CCSDS Informational Report that presents an overview of a concept for a Mission Operations Service Framework for use in spacecraft monitoring and control.

In this context, MO refers to end-to-end services between functions, based on the ground or even resident onboard a spacecraft, which are responsible for mission operations.

Standardization of an MO Service Framework offers a number of potential benefits for the development, deployment, and maintenance of mission operations infrastructure:

- increased interoperability between agencies, at the level of spacecraft, payloads, or ground-segment infrastructure components;
- standardization of infrastructure interfaces, even within agencies, leading to re-use between missions and the ability to establish common multi-mission infrastructure;
- standardization of operational interfaces for spacecraft from different manufacturers;
- reduced cost of mission-specific deployment through the integration of reusable components;
- ability to select the best product for a given task from a range of compatible components;
- greater flexibility in deployment boundaries: functions can be migrated more easily between ground-segment sites or even from ground to space;
- standardization of a limited number of services rather than a large number of specific inter-component interfaces;
- increased competition in the provision of commercial tools, leading to cost reduction and vendor independence;
- improved long-term maintainability, through system evolution over the mission lifetime through both component and infrastructure replacement.

The subjects covered in this volume include:

- Approach to Service Identification;
- Service Structure;
- Mission Operations Functions;
- Identified Mission Operations Services;
- Mission Operations Service Concept;
- Reference Model;
- Message Abstraction Layer (reference [1]);
- Common Object Model;
- Service Specifications;
- Language API.

4.3 MISSION OPERATIONS SERVICES MESSAGE ABSTRACTION LAYER

The Mission Operations Services Message Abstraction Layer Recommended Standard (reference [1]) defines the Mission Operations Message Abstraction Layer (MAL) in conformance with the service framework specified in Mission Operations Services Concept (reference [3]).

These Mission Operations Services are defined in terms of the MAL. It should be noted that the terms ‘consumer’ and ‘provider’ refer to the service, not the message itself. The sender of a message is ‘consuming’ the service that is ‘provided’ by the receiver of the message.

- The *Send* pattern is the basic interaction of which all other patterns can be considered extensions. It is the simple passing of a message from a consumer to a provider. Because there is no message ‘conversation’ implied with a simple *Send*, there is no requirement for a transaction identifier in the message. However, one may be specified. No return message is sent from the provider to the consumer, so the consumer has no indication the provider has received the message. The *Send* pattern is expected to be used for non-critical messages where the possible loss of one or more of these messages is not considered critical. An example would be regular heartbeat-type messages.
- The *Submit* pattern extends the *Send* pattern by providing a return acknowledgment message from the provider back to the consumer. The service specification details the meaning of the acknowledgment message for a specific operation. The *Submit* pattern is used for simple operations that complete quickly but must be confirmed to the consumer.
- The *Request* pattern provides a simple request/response message exchange. Unlike the *Submit* pattern, no acknowledgement is sent upon reception of the request; however, a data response is sent. The lack of an acknowledgement and only the return

- data response for this pattern means that it is primarily expected to be used for situations where the operation takes minimal time. It is expected that the *Request* pattern is to be used only for operations that complete in a relatively short period of time. If a more extended or indeterminate period is possible then the more advanced *Invoke* or *Progress* patterns should be specified.
- The *Invoke* pattern extends the *Request* pattern with the addition of a mandatory acknowledgement of the initial message. The *Invoke* pattern is expected to be used when there is a significant or indeterminate amount of time taken to process the request and produce the data response. The provision of the service-defined acknowledgement message allows an operation to return supplementary, status, or summary information about the request before processing continues (for example, an identifier used for querying *Invoke* status using another operation).
 - The *Progress* pattern extends the *Invoke* pattern with the addition of a set of mandatory progress messages. The type of progress messages and their number is defined by the service and not by the pattern. The *Progress* pattern is expected to be used when there is a significant or indeterminate amount of time taken to process the request and produce the data response, and where monitoring of the progress of the operation is required or the data response is to be returned in blocks.
 - The *Publish-Subscribe* pattern for both the consumer and provider register/deregister has a predefined pattern message structure that allows an implementation of the message broker to manage the mapping of consumers to updates and hides this complexity from provider applications.

4.4 MISSION OPERATIONS SERVICES COMMON OBJECT MODEL

The Mission Operations Services Common Object Model Recommended Standard (reference [7]) defines the Mission Operations Common Object Model (COM) in conformance with the service framework specified in annex B of Mission Operations Services Concept (reference [3]).

The MO COM is a generic service template that provides a Common Object Model to the Mission Operation services. These Mission Operations Services are defined in terms of the COM and the MAL (reference [1]).

The services described in this volume include:

- An *event* is a specific object representing ‘something that happens in the system at a given point in time’. The event service defines a single publish-subscribe operation that supports the monitoring of events generated by other components. An *event*, as it is a COM object, is identified by the normal object fields (domain, object type, and object instance identifier) with the addition of a string name. The name provides a more human friendly means to identify the *event*.

- The *Archive* service provides a basic archiving function for COM objects. It follows the Create Retrieve Update Delete (CRUD) principles and allows simple querying of the *Archive* (more complex queries are supported but the specifications of these are outside this the scope of the standard). As changes are made during the lifetime of an object, this information is distributed to consumers using the service defined operations; as long as these updates follow the COM standard for object identification they can also be stored in a COM archive. By storing these updates in an *Archive*, any historical replay and retrieval functions can correctly reflect the history of the objects.
- The *Activity Tracking* service, or Activity Service for short, provides the ability to track the progress of activities; an activity is anything that has a measurable period of time (a command, a remote procedure, a schedule, etc.). The basic service provides the ability to track the progress of MAL operations, but it is expected to be used for other processes where appropriate. It defines an event pattern that supports the reporting of the progress of activities from the initial consumer request, tracking its progress across a transport link, to reception by the provider and execution in that provider. The service uses the event service to report the progress of activities which supports the concept of external monitoring where one component is able to monitor the activities in the system without requiring knowledge of what components are active. This permits the implementation of a single component for the monitoring of activity in the system and also for the archiving of this activity. It also supports monitoring of activities that are passed via a chain of components to a provider; these intermediate components are referred to as relays.

4.5 MISSION OPERATIONS SERVICES COMMON SERVICES

The Mission Operations Common Services (reference [8]) Recommended Standard defines Mission Operations Common Services in conformance with the service framework specified in Mission Operations Services Concept (reference [3]).

The Common Services are a set of services that provide support facilities to the Mission Operation Services. These services are defined in terms of the COM and the MAL.

The services described in this volume include:

- The *Directory* service provides publish and lookup facilities to service providers and service consumers. It allows providers to publish their location in the form of a URI (Universal Resource Indicator) so that consumers can locate it without having to know in advance the location. Strictly speaking a *Directory* is not required if a well-known service is to be used; however, in most circumstances a *Directory* provides required flexibility in the location of services. The *Directory* is organized in a hierarchical tree structure of directory Nodes. Each Node is identified by a domain, network zone, and session and contains a list of services that are currently available at that node. Alternate networks and sessions are supported by having the same domain identifier but using different network or session identifiers, for example, for a specific domain there may be several sessions in the *Directory* service and each of those may have a different set of service providers.

- The *Login* service allows an operator to provide authentication information to the system. It takes the operator's credentials and uses a deployment-specific mechanism to authenticate the operator; the result of this is used by the MAL during access control. The *Login* service and the access control provided by the MAL are fully dependent on a deployment-specific security architecture (for example the authentication protocol Kerberos). Both layers (Common and MAL) provide access to, and use of, this security service. Neither layer implements this itself. (See Mission Operations Message Abstraction Layer (reference [1]) for more information regarding access control.)
- When a component of the system requires some input from an operator that is not as a *result* of an action of the operator, for example, confirmation of a critical automated action, the *Interact* service allows a component to request that information from an appropriate operator.

Four types of operator interaction are supported:

- The *Acknowledge* interaction is used when the Operator is required to acknowledge an operation (for example, a simple 'Ok'-style interaction).
 - The *Confirm* interaction is used when the Operator is required to confirm or decline an operation (for example, a 'Yes/No'-style interaction).
 - The *Choice* interaction is used when the Operator is presented with a list of options to choose from (for example, a drop-down list of choices).
 - The *Get values* interaction is used when the Operator is required to provide a set of values (for example, value entry text fields).
- The *Configuration* service provides the ability to transfer the configuration of a service to and from a service provider. It provides facilities for the management of configurations held by a provider if applicable to that implementation. Implementations of the *Configuration* service may support bespoke configuration upload mechanisms (such as file upload). This is supported by the *Configuration* specification where consumers would be notified of a new configuration; however, the details of these upload methods are outside of the scope of the Recommended Standard.
 - A historical *replay* is considered to be a session distinct from other active sessions such as the current live session or simulation sessions. Each *replay* session is created based on an existing live or simulation session (limited to a specific domain and zone); the *replay* session is a read-only copy of the base session that can be ' browsed ' under consumer control. The *replay* service allows a consumer to create, control, and delete *replay* sessions. Once a *replay* has been created, a consumer can:
 - Single-step through history forwards and backwards;
 - Play the history forwards and backwards;
 - Adjust the replay rate of the replay session;
 - Delete the replay session.

4.6 MISSION OPERATIONS SERVICES MONITOR AND CONTROL

The Mission Operations Monitor and Control Services (reference [7]) Recommended Standard defines Mission Operations Monitor and Control (M&C) services in conformance with the service framework specified in the Mission Operations Services Concept Informational Report (reference [3]).

The M&C services are a set of services that enables a mission to perform basic monitoring and control of a remote entity. These services are defined in terms of the Mission Operations Common Object Model Recommended Standard (reference [7]) and the Mission Operations MAL (reference [1]).

The services described in this volume include:

- The *Action Service* enables consumers to control the remote system, typically a spacecraft; however, there is no restriction on what the remote system may be. *Action* invocation operations include issuing of action directives by an authorized client to the remote service provider, and the subsequent monitoring of the evolving execution status of that action by both the initiator and other client functions.
- *Parameter* status monitoring is performed by publishing the status of a set of predefined monitoring parameters that contain status information. Monitoring parameters have an evolving status represented by a chronological sequence of status updates over an unbounded lifetime. Status updates may be periodic, change-based, or a mixture of the two. Monitoring parameters are basic types such as strings or integers. Composition of parameters is supported by defining an appropriate aggregation.
- *Alerts* are raised asynchronously to report a significant event or anomaly. Alerts may originate within the remote service provider (spacecraft or other controlled system) itself or within an associated ground-based component in response to a transition in some monitored status. Alerts are characterized by an identifier and a set of arguments. However, it is possible that some systems require text-only alert messages where the body of the alert is a free-form text message. For these systems it is possible to define a single alert definition that contains a single string argument; however, it should be noted that translation software shall be required when moving argument-based messages and string-based messages between the two formats.
- The *Check* service allows the consumer to define a set of checks to be applied to parameters and then periodically sample the values of those parameters and check them. If a check is violated (for example, goes outside of the specified limits), the consumer is notified by the generation of a COM event. In addition to this, the service provider maintains a list of parameters currently violating any associated checks. Each check has an associated severity level. A minimal list of check types is supported; however, service implementations may support additional custom types. The following is the list of checks that must be supported:
 - a *Limit* checks that the value lies within a specified range;

- an Expected value checks that the value is checked against a specified value or value of another parameter;
 - a Delta checks that the change in value is checked against a pair of thresholds.
- The *Statistics* service allows the consumer to associate parameters to defined statistical evaluations (e.g., *min*, *max*, *mean*, *standard deviation*) and periodically sample the values of those parameters and evaluate them. The resultant statistics evaluations are provided to consumers who register interest in them. The service allows the set of parameters associated with a statistic to be added to and deleted from; however, it does not allow new statistic algorithms not currently supported by a service provider to be defined. For example, if a service provider supports *min* and *max* statistics, it is possible to add parameters to these statistical functions, but it is not possible to add a new statistic ‘standard deviation’, as this is not supported by the service provider.
- Another logical extension to basic monitoring is data aggregation. This *Aggregation* service provides the capability to acquire several parameter values in a single request. The aggregation might be one of the following:
- predefined by the service provider, e.g., housekeeping parameters;
 - predefined at runtime by the consumer, e.g., a diagnostic report.

For example, the user may request a data product that comprises the accelerations and angular rates of the spacecraft. This would be acquired by reading the appropriate gyros and accelerometers onboard and returning the data as a set. However, if the service provider is actually on the ground, for example, a spacecraft kernel acting as a proxy, then it may collate the set from its current ‘state vector’ of all reported parameters.

- A functional extension of the other services is to add the engineering unit conversion capability. For example, using this *Conversion* service with the parameter service, when consumers register for updates for a specified parameter, they receive a value that represents the parameter being read, for example, a quantized temperature value with a unit rather than the raw data value.

5 DOCUMENT ROADMAP

5.1 INTRODUCTION

This section describes an approach to developing a proposed Recommended Standard for telerobotic operations.

5.2 TESTING AND PROTOTYPING

This subsection describes the specific methods that participating agencies will use to look for opportunities to perform early tests of proposed Recommended Standard prototypes.

Participating organizations should look for opportunities to involve their technology development laboratories in agency field-testing as a way to provide opportunities for interoperability testing. Although directly supporting the development of a common communications network infrastructure between participating agencies is not anticipated, the development of such infrastructure is encouraged as a way of reducing the costs associated with the independent prototyping and test activities required of CCSDS standards development groups.

To encourage the widest possible test coverage, prototypes are encouraged in different computer languages, such as C++ and Java, and testing should occur against both actual and virtual robotic assets. It is expected that each official prototyping effort will implement all defined messages and APIs; however, because of the effort required, individual prototypes may implement simplified versions of the specified services. When taken together, all services will receive at least one complete implementation within the full set of proposed Recommended Standard prototypes.

5.3 ELIMINATING OVERLAP WITH OTHER STANDARDS

A proposed Recommended Standard for telerobotic operations should neither compete with nor duplicate existing CCSDS standards.

5.4 SELECTION OF REFERENCE IMPLEMENTATION LANGUAGE

In an effort to continuously test and improve upon the emerging proposed Recommended Standard for telerobotic operations, a language should be selected in which to create implementation of the specification to be used as a definitive interpretation for the specification. This Reference Implementation should be developed concurrently with the specification and test suite so as to verify that the specification is implementable and testable. The Reference Implementation would be the standard against which other implementations will be measured and would help to clarify intent of specification where other sources of information are lacking.

It is anticipated that the selected reference implementation language will be either Java or C++.

6 OTHER INDUSTRY STANDARDS

6.1 OVERVIEW

There exist a number of relevant robotics standards that should inform development of a proposed Recommended Standard. Short descriptions of these standards are included in this section; the referenced standard itself should be consulted for a complete description.

6.2 JOINT ARCHITECTURE FOR UNMANNED SYSTEMS

Originally developed by the United States Department of Defense in 1998, JAUS is an open architecture for the domain of unmanned systems, featuring vehicle platform independence, mission isolation, computer hardware independence, technology independence, and operator use independence.

The JAUS standards are owned and developed by the Society of Automotive Engineers (SAE) under the Aerospace Standards Unmanned Systems Steering Committee (AS-4). All of the standard documents that define JAUS can be purchased online directly from SAE.

JAUS includes several standards specification documents, including network transport, interface definition language, core services, mobility services, human machine interface services and manipulation services. JAUS uses a Service Oriented Architecture (SOA), and is therefore fundamentally compatible with the SOA-based CCSDS Mission Operations Services. The JAUS standard focuses on the operation of autonomous ground, air and undersea vehicles, but does not address space-based operations or human-robot collaboration.

Several implementations of the JAUS standard exist, including OpenJAUS, The JAUS Toolset (JTS), and JAUS++. OpenJAUS is a commercial product for non-academic uses, while JTS is a JAUS-compliant system design tool. JAUS++ is an Open Source implementation of many of the JAUS services in C++. There do not seem to be any implementations of the JAUS standard in Java or other languages, although JTS can emit Java code as part of its code-generation process.

The JAUS service set is well aligned with the service set applicable to a proposed Recommended Standard for telerobotic operations, work on which should include analysis of the JAUS message set to ensure that the telerobotic operations message set is not necessarily incompatible at a conceptual level.

6.3 ROBOT APPLICATION PROGRAMMING INTERFACE DELEGATE

The Robot Application Programming Interface Delegate (RAPID) is a set of software data structures and routines that simplify the process of communicating between multiple diverse robots and their command and control systems. RAPID is not intended to be an all-encompassing API for robot communication, but rather it is a compatibility layer that permits

tools and robotic assets to exchange data and information and allows operators to communicate with heterogeneous robots in a uniform way. RAPID is a compatibility layer that delegates information between robots that speak different languages.

The RAPID specification includes definitions and APIs for messages and services that support collaborative, supervisory telerobotic operations over near-Earth time delay. RAPID is not a middleware specification, although safety and time-delay capabilities do imply requirements on implementing middleware systems. As currently implemented, the RAPID system can be considered a software reference implementation for remote operations.

RAPID has been in development and use within NASA since 2007, and current development and test is supported by the Space Technology Mission Directorate through the Game Changing Development Program's Human-Robotic Systems Project and Human Exploration Telerobotics Project.

6.4 COUPLED-LAYER ARCHITECTURE FOR ROBOTIC AUTONOMY

CLARATy, which stands for Coupled-Layer Architecture for Robotic Autonomy, is an integrated framework for reusable robotic software. It defines interfaces for common robotic functionality and integrates multiple implementations of any given functionality. Examples of such capabilities include pose estimation, navigation, locomotion and planning. In addition to supporting multiple algorithms, CLARATy provides adaptations to multiple robotic platforms.

CLARATy is a domain-specific robotic architecture designed with four main objectives:

- to promote the reuse of robotic software infrastructure across multiple NASA-related research efforts;
- to promote the integration of new technologies developed by the robotics community onto rover platforms;
- to mature robotic capabilities through reuse and enable independent formal validation; and
- to share the development with the robotic community to promote rapid advancement and leveraging of capabilities.

CLARATy differs from a proposed Recommended Standard for telerobotic operations in that CLARATy addresses primarily the integration of heterogeneous robotic control technologies onboard robotic agents, while a proposed Recommended Standard for telerobotic operations emphasizes the operations integration of disparate sets of human and robotic agents. One could consider the development of a CLARATy module that would support the integration of any robotic platform that is using CLARATy as its operating software with robotic platforms using other robotic operating systems. The common telerobotic operations command and control regime that integrates the operation of these agents is the subject of a proposed Recommended Standard for telerobotic operations.

The CLARAty Web site, from which much of the material in this section was adapted, should be consulted for additional information.

6.5 SENSOR MODEL LANGUAGE

The Open Geospatial Consortium’s Sensor Model Language (SensorML) (reference [12]) provides “a robust and semantically-tied means of defining processes and processing components associated with the measurement and post-measurement transformation of observations. This includes sensors and actuators as well as computational processes applied pre- and postmeasurement.” (reference [12], ix)

Like CCSDS, the Open Geospatial Consortium is an international standards organization with membership from the space community. The SensorML standard is an outgrowth of OGC’s Sensor Web Enablement (SWE) activity, which sought to “enable a ‘Sensor Web’ through which applications and services will be able to access sensors of all types, and observations generated by them, over the Web.” (reference [12], xii)

The SensorML effort should inform development of a proposed Recommended Standard for telerobotic operations definition effort with respect to enabling the exchange of operations-enabling science data products via the Data Product Service.

6.6 ROBOT OPERATING SYSTEM

The Robot Operating System (ROS) is a set of software libraries and tools for building robot applications. ROS features a rich set of hardware drivers, functional modules, and developer tools, and is popular in both the educational and research realms. ROS is being used on a number of advanced space telerobotics projects, including ISS Astrobee and Robonaut 2. ROS provides a standard communications infrastructure in the form of a message-passing system upon which has grown a large set of standard messages covering common use cases in terrestrial robotics. ROS’s developer tooling includes robot description languages, visualization and simulation components and diagnostic tools.

While ROS is a very popular robotics toolkit, it does not address the special needs of the safe, collaborative operation of mixed teams of human and robotic assets in the exploration of space. ROS does not address high-latency operations or operations over disruption-prone communications networks. However, the effort to promote international standards in telerobotic operations can benefit from a close assessment of the standard ROS robot messages and adopt relevant design principles from the efforts of the ROS community.

ANNEX A

LEXICON

This Lexicon contains definitions of select terms and instructions on how to render terms contained in this Report into other languages and/or ontologies. This Lexicon is intended to document decisions related to the selection of the normative terminology. This Lexicon will hopefully eliminate use of alternative textual references in the main body of the Book and serve as a historical record for lexical decisions made during the editorial process.

Additional lexicons or annexes may be created to offer more in-depth comparisons between participating organizations' approaches to collaboration, or as corporate knowledge bases for other significant discussions and/or findings made during Report development. Definitions from the SANA Registry of CCSDS Terms are marked '(CCSDS Term)'. Definitions common to the telerobotic operations field are marked '(Term of Art)'.

action: An atomic (non-interruptible) control directive of mission operations (equivalent to a telecommand or ground-segment directive) that can be initiated by manual or automatic control sources, via the M&C service. An action may have arguments and its evolving status can be observed. An action can typically apply pre- and post-execution checks. (CCSDS Term)

activity: An automated mission operations function; typically an operations procedure, batch task or other software process. An activity can be individually scheduled or initiated. In principle, an activity is non-atomic, has duration, and can be controlled once initiated. An activity may have arguments and its evolving status can be observed. An activity may generate multiple actions, and its behavior can be dependent on status observed at runtime. (CCSDS Term)

Agent: An 'Agent' is an entity that can be monitored or commanded telerobotic operations services. When referring to a robot or class of robots, the term 'Robot' can be used in place of 'Agent'. When referring to a human or class of humans, the term 'Astronaut' or 'Operator' can be used in place of 'Agent', depending on the situation. (Term of Art)

alert: An asynchronous notification, such as a non-nominal event, of significance to mission operations. Alerts may be used to notify such events to operators, initiate an automatic response, or synchronize asynchronous functions. Alerts may have arguments. (CCSDS Term)

application programmers' interface: The definition of the exposed or 'public' interface to a software component that can be used by another software component. In a Service Oriented Architecture context, an API corresponds to a language- or technology-specific implementation of an abstract service specification. This constitutes the code classes, types and functions utilized by a programmer when implementing the service provider and service consumer. (CCSDS Term)

argument: A run-time parameter provided to various control items on invocation, e.g., telecommand arguments. Arguments apply to actions, activities and alerts among other items. (CCSDS Term)

bilateral control: A system of teleoperation control in which control signals flow from the operator to the robotic agent to carry out a task, while at the same time, feedback signals flow from the robotic agent to the operator, usually to provide an indication of the forces experienced by the robotic agent as a method to improve the operability of the system. (Term of Art)

capability: A capability is a core ability of a system, or a function offered by a service. (CCSDS Term)

consumed service interface: The API presented to the consumer component that maps from the service operations to one or more service data units contained in MAL messages that are transported to the provided service interface. (CCSDS Term)

domain: A namespace that partitions separately addressable entities (e.g., actions, parameters, alerts) in the space system. The space system is decomposed into a hierarchy of domains within which entity identifiers are unique. (CCSDS Term)

event: A time-stamped message, containing (changes in) information about information objects associated with a service, which is exchanged across service interfaces and potentially stored in the service history. (CCSDS Term)

management service interface: A service interface that exposes management functions of a service function contained in a component for use by service consumers. (CCSDS Term)

Mission Operations Services: A suite of end-to-end application-level services that constitute a Service Oriented Architecture for space mission operations. (CCSDS Term)

operation: In object-oriented programming, a method, function, or message defined for a class of objects; in the Mission Operations Services context, a control primitive that can be performed across the service interface. (CCSDS Term)

parameter: An item of mission operations status information that can be individually subscribed to by a service consumer, via the M&C service. A parameter has multiple attributes, including: raw value, engineering value, validity, check status and (optionally) statistics. (CCSDS Term)

provided service interface: A service interface that exposes the service function contained in a component for use by service consumers. It receives the MAL messages from a consumed service interface and maps them into Application Program Interface (API) calls on the provider component. (CCSDS Term)

proxy: A proxy function or component is one that acts locally in the place of a remote service provider, such as a spacecraft. There is a proxy function for each service. It provides

a dual role. Firstly, it provides a permanent point of contact for service consumers where the link to the remote service provider is intermittent, maintaining an image of current status, buffering operations and managing the service history. Secondly, it can act as an isolation layer and adapter to actual protocols employed on the space-ground interface. (CCSDS Term)

replay: The act or interface associated with viewing data from a service history in the same manner as live operation. Service events are dynamically replayed over an evolving time period. (CCSDS Term)

retrieval: The act or interface associated with of withdrawing a data set by a time range from a service history. Retrieval is mainly intended for fast access to a block of service history for display of data trends or logs over a period of time, or to be used in analytical tasks. (CCSDS Term)

service: A set of capabilities that a component provides to another component via an interface. A service is defined in terms of the set of operations that can be invoked and performed through the service interface. Service specifications define the capabilities, behavior and external interfaces, but do not define the implementation. (CCSDS Term)

service consumer: A component that consumes or uses a service provided by another component. A component may be a provider of some services and a consumer of others. (CCSDS Term)

service data unit: A unit of data that is sent by a service interface and is transmitted, semantically unchanged, to a peer service interface. (CCSDS Term)

service history: The operational data archive for a service. This is the data required to reconstitute a historical view of information at the service interface, either using replay or retrieval access methods. It corresponds to the persistent sequence of all service events over a period of time, to which a service consumer could have subscribed. Examples of service histories include parameter history, action history and alert history. Alternative implementations are possible, based on archiving of protocol messages (e.g., packets) and re-processing. (CCSDS Term)

service interface: A set of interactions provided by a component for participation with another component for some purpose, along with constraints on how they can occur. A service interface is an external interface of a service where the behavior of the service provider component is exposed. Each service will have one defined provided service interface, and may have one or more consumed service interfaces and one management service interface. (CCSDS Term)

Service Oriented Architecture: Service Oriented Architecture is a paradigm for organizing and utilizing distributed capabilities that may be under the control of different ownership domains. It is not itself a solution to domain problems but rather an organizing and delivery paradigm that enables one to get more value from use both of capabilities which are locally 'owned' and those under the control of others. (Term of Art) (reference [13])

service provider: An application or component that offers a service to another by means of exposing a service interface, while hiding details of its implementation. (CCSDS Term)

session: A session defines the time frame for a service. A session may be live or historical, real or simulated. A service consumer may join any existing session by subscribing to a service for that session. Within a given system there may be multiple concurrent sessions, to support simulated and/or historical replay sessions in parallel with live operations. Within service history there may be multiple session histories, corresponding to live operations and simulated sessions. (CCSDS Term)

software component: A software unit containing the business function. Components offer their function as services, which can either be used internally or which can be made available for use outside the component through provided service interfaces. Components may also depend on services provided by other components through consumed service interfaces. (CCSDS Term)

ANNEX B

ABBREVIATIONS

The [SANA CCSDS Abbreviations registry](#) should be consulted for a complete list of official CCSDS abbreviations.

AMS	CCSDS Asynchronous Message Service
API	application programmers' interface
CCSDS	Consultative Committee for Space Data Systems
CFDP	CCSDS File Delivery Protocol
CLARAty	Coupled-Layer Architecture for Robotic Autonomy
DDS	data distribution service
DOF	degrees of freedom
DTN	Delay-/Disruption-Tolerant Networking
J AUS	Joint Architecture for Unmanned Systems
LEO	low earth orbit
MAL	Message Abstraction Layer
MO	Mission Operations
NEO	near-earth object
OGC	Open Geospatial Consortium
RAPID	Robot Application Programming Interface Delegate
RTI	Real-Time Innovations
SANA	Space Assigned Numbers Authority
SM&C	CCSDS Spacecraft Monitor & Control
SOA	Service Oriented Architecture
SWE	Sensor Web Enablement
UDP	User Datagram Protocol