CCSDS compliant Ground Network Schedule Interfaces from an ESA perspective

Holger Dreihahn [1], Colin Haddow[2] and Martin Unal[3]
ESA/ESOC Robert-Bosch Str.5, D-64293 Darmstadt, Germany

For many envisaged and flying space missions cross support plays an integral role in mission design and mission operations. Thanks to the CCSDS space link (SL) and space link extension (SLE) recommendations and others, spacecraft and mission control centers of one agency can interoperate with ground stations of other space agencies and commercial or governmental organizations. However, when it comes to planning and scheduling of the ground network elements involved in cross support, no standardized interfaces and approaches that have been implemented operationally exist yet.

Having recognized this situation, CCSDS is developing interfaces to support planning and scheduling of ground stations in the context of cross support. The aim is to reduce the labor intensive task of defining the planning and schedule interfaces over and over again on a per mission basis between the interoperating partners. Finally the recommended standards shall reduce the number of private interfaces and shall facilitate standard compliant implementations. CCSDS is defining the integral elements for these planning and scheduling interfaces, initially concentrating on defining the required data formats: To exchange scheduling information related to ground stations and / or relay satellites, the CCSDS Simple Scheduling Format Specification (CCSDS 902.1-R-1) was prepared, has been under agency review in 2015 and is currently being finalized for publication. As a counterpart to this work is presently underway on defining the data entity required for the submission of planning and service requests with the intention that this also becomes a CCSDS standard.

In this paper we will introduce the two CCSDS planning and schedule interface specifications and set them into context. Looking from an ESA perspective we will show the envisaged adoption of these planning and scheduling specifications at ESA and will demonstrate the envisaged high level architecture of the foreseen implementation in the context of the ESTRACK Management System, ESA's planning and scheduling system of the ESTRACK ground network.

Nomenclature

CCSDS	=	Consultative Committee for Space Data Systems
EMS	=	ESTRACK Management System
ESTRACK	=	ESA Space Tracking Network
SL	=	Space Link
SLE	=	Space Link Extension

[1] Technical Officer, ESA/ESOC Ground Systems Engineering Department, Robert-Bosch Str.5, D-64293 Darmstadt, Germany
[2] Design Authority, ESA/ESOC Ground Systems Engineering Department, Robert-Bosch Str.5, D-64293 Darmstadt, Germany
[3] Ground Operation Manager, ESA/ESOC Operations Department, Robert-Bosch Str.5, D-64293 Darmstadt, Germany

I. Introduction

The need of scheduling external ground network support exists for space missions today more than ever. In this paper we will focus on the scheduling aspect of ground network resources in a CCSDS compliant way. The main elements of this are the CCSDS Simple Schedule as specified by [1] currently being finalized, which defines a format which allows a ground network to publish booked passes in a standardized way. For each pass the used resources such as the antenna is provided. Furthermore the services provided during the pass as well as the service parameterization are published.

To request passes at a potentially external ground network, the CCSDS Service Management Utilization Request Formats as specified by [2] currently being drafted provides the means to request of passes and the required services.

In CCSDS terminology the Utilization Manager is in charge of negotiating the resources to conduct mission operations. In the context of cross support scheduling this translates to negotiating passes, or in CCSDS terminology Scheduled Packages, with an external ground network. Within the external ground network the CCSDS term Complex Manager denotes the entity in charge of scheduling the available resources.

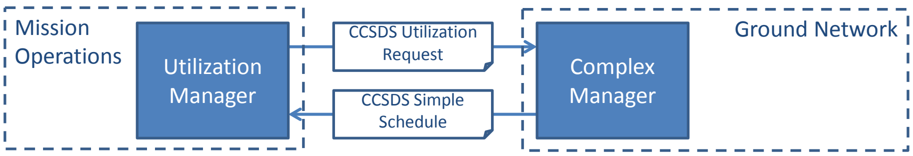

Figure I-1 Basic Elements of Cross Support Scheduling

With the introduction of the CCSDS Simple Schedule [1] and Utilization Requests [2], CCSDS provides the building blocks to build CCSDS compliant schedule interfaces. It must be noted that these CCSDS compliant schedule interfaces *can be built without* a full implementation of CCSDS Service Management [3] (or its successor), but there are elements to integrate with CCSDS Service Management if this is required, for instance at a later stage in time. In the following we will discuss some history of relevant CCSDS standards and discuss on-going future directions. Then we will have a detailed look at the CCSDS Simple Schedule [1] and Utilization Requests [2], before we present ESA's approach of adopting and implementing a CCSDS compliant schedule interface.

II. Related CCSDS Work and bit of History

In 2009 CCSDS has published the first Space Communication Cross Support Service Management Blue Book 1 (SCCS SM B-1) [3]. In a nutshell SCCS SM B-1 specifies four services, see also [4]:

- Service Agreement Service – *Contains general agreement details and limits for configuration profile values*
- Configuration Profile Service – *Management of configuration data for the RF and communication equipment*
- Trajectory Prediction Service – *Deals with management of predicted trajectory data*
- Service Package Service – *Provides the means for allocation of passes*

While the first three services deal with configuration and trajectory data, the last one, the Service Package Service, basically deals with cross support booking of passes in the form of Service Packages. However, while the Service Package Service offers functions to request, modify, cancel and query Service Packages, it does not have an equivalent to the CCSDS Simple Schedule providing a larger view on the scheduled Service Packages of a network. Furthermore the SCCS SM B-1 Service Package Service has dependencies to all other SCCS SM B-1 services, although [3] defines a minimum compliance level with four Service Package service operations. As pointed out by [4], the implementation of all SCCS SM B-1 services typically affects quite a number of different organizational units within a space agency or a commercial network provider and is not trivial. At the example of ESA/ESOC [4] shows that SCCS SM B-1 service implementation would require support of Scheduling Office, ESTRACK Operations and Flight Dynamics.

In contrast to SCCS SM B-1, the CCSDS Simple Schedule [1] and Utilization Requests [2] have optional dependencies to Service Management allowing the implementation of a CCSDS compliant schedule interface without implementation of Service Management, while an implementation of Service Management can be used or added at a later stage in time.

III. CCSDS Simple Schedule Format and Extensions

According to [1] the ‘Simple Schedule Format Recommended Standard specifies a standard format for use in transferring scheduling information related to apertures at ground stations and/or relay satellites between space agencies and commercial or governmental spacecraft operators’

For that purpose [1] defines an XML exchange data format, the structure is shown in Figure III-1.

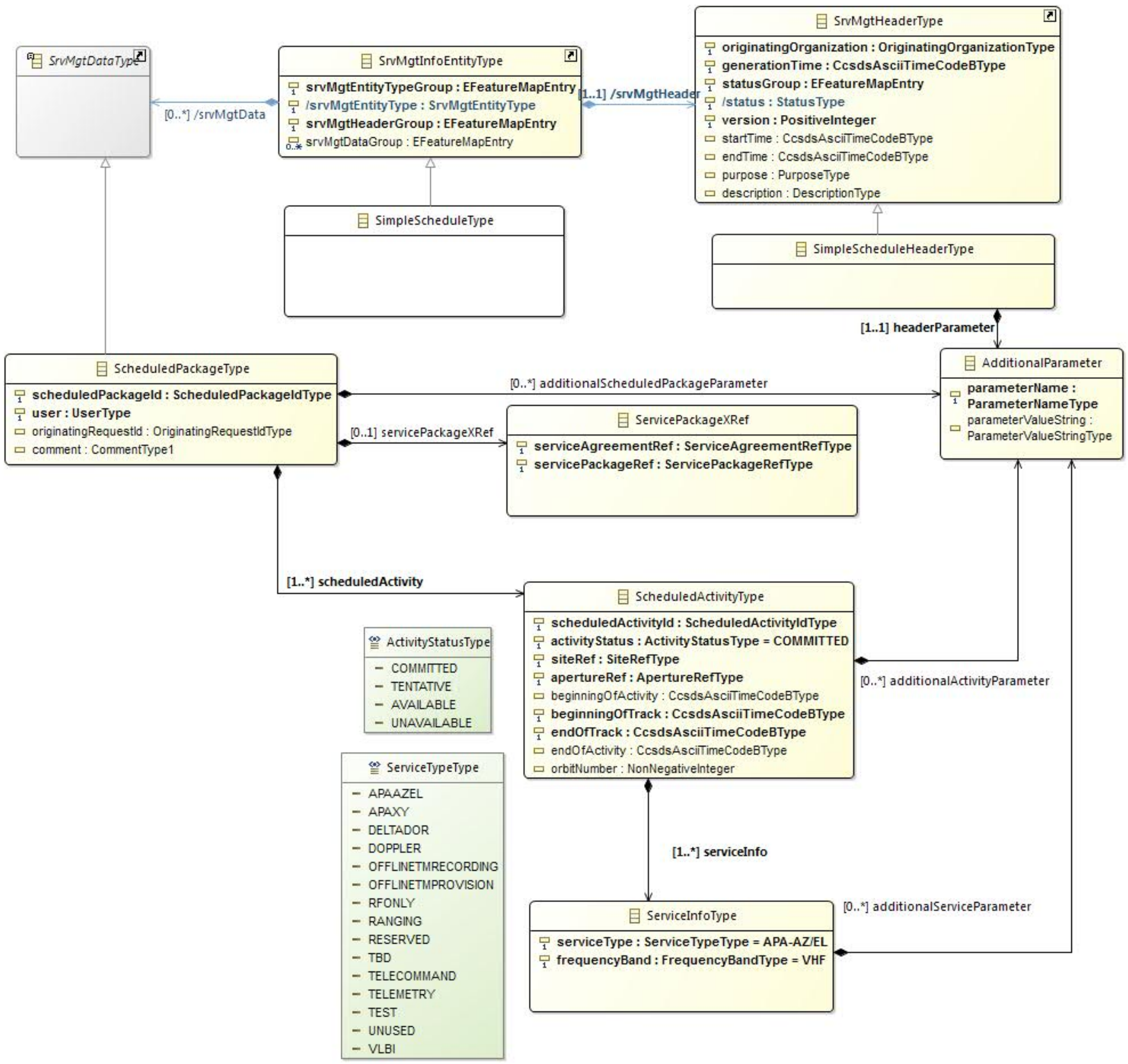

Figure III-1 CCSDS Simple Schedule Format

Based on the general Service Management types **SrvMgtDataType**, **SrvMgtInfoEntityType** and **SrvMgtHeaderType**, the Simple Schedule specific types are defined. The **SimpleScheduleHeaderType** specializes the general type and adds the capability to add header parameters as **AdditionalParameter**. The Simple Schedule uses the concept of **AdditionalParameter**s at several places to allow extensions for information which cannot be specified up-front.

The content of the Simple Schedule are instances of the **ScheduledPackageType**, provide a grouping for **ScheduledActivityType** instances belonging to the same scheduled package, as well as an optional **ServicePackageXRef** allowing the reference of service packages and the corresponding service agreement.

Every **ScheduledActivityType** specifies one or more services actually provided during the period bounded by **ScheduledActivityType/beginningOfTrack** and **ScheduledActivityType/endOfTrack.** Examples of provided services include, but are not limited to TELECOMMAND services, TELEMETRY services or DELTADOR services.

Each simple schedule has a validity period defined by the start and end time of the header. Reference [1] mandates that each simple schedule shall contain all activities somehow overlapping the schedule validity. The rationale here is that a schedule must provide a complete picture of the covered period, even if this implies that two or more schedules covering different periods can contain duplications of activities. Example: Imagine two schedules covering two consecutive weeks, each schedule covers one week. A long running maintenance activity starting before the first schedule and ending after the second schedule will be included in both schedules. This places on the systems processing the schedules the burden of handling and potentially eliminating the duplicated activities. However, also alternative definitions like the common ‘a schedule contains only activities starting in the schedule validity period’ have drawbacks. The most prominent problem for such a definition is the lack of an indicator to which point in time one needs to go back to obtain all activities overlapping a specific period in question. Another way of addressing the problem sketched out above would be the notion of schedule boundary conditions providing this information, but this concept has been dropped in favor of simplicity.

Other potential future Simple Schedule extensions under discussion include

- Association of Scheduled Packages to address MSPA and Delta Door scenarios
- Filters to state which filtering has been applied to create the schedule
- Inclusion of a file name to address cases where the schedule is stored or transferred as a file and has to be renamed

While some of these changes can be covered by making use of the **AdditionalParameter** elements, others like the Association of Scheduled Packages may require new types.

IV. CCSDS Service Management Utilization Request Formats (SMURF)

As specified by [2], ‘the Service Management Utilization Requests Formats forms the basis from which various types of requests required by Service Management are derived. These can be categorized into 5 types’:

1. Planning Information Request
2. Report Request
3. Information Request
4. Submission Request
5. Service Package Request

In the context of this paper the Report Requests are relevant because they allow to request Simple Schedules.

Furthermore we will cover the Service Package Request, which allows to request

- New Service Packages
- Updates for existing Service Packages
- Delete Service Packages
- Select alternatives of Service Packages

The overall structure of the request subset as specified by [2] is depicted in Figure IV-1.

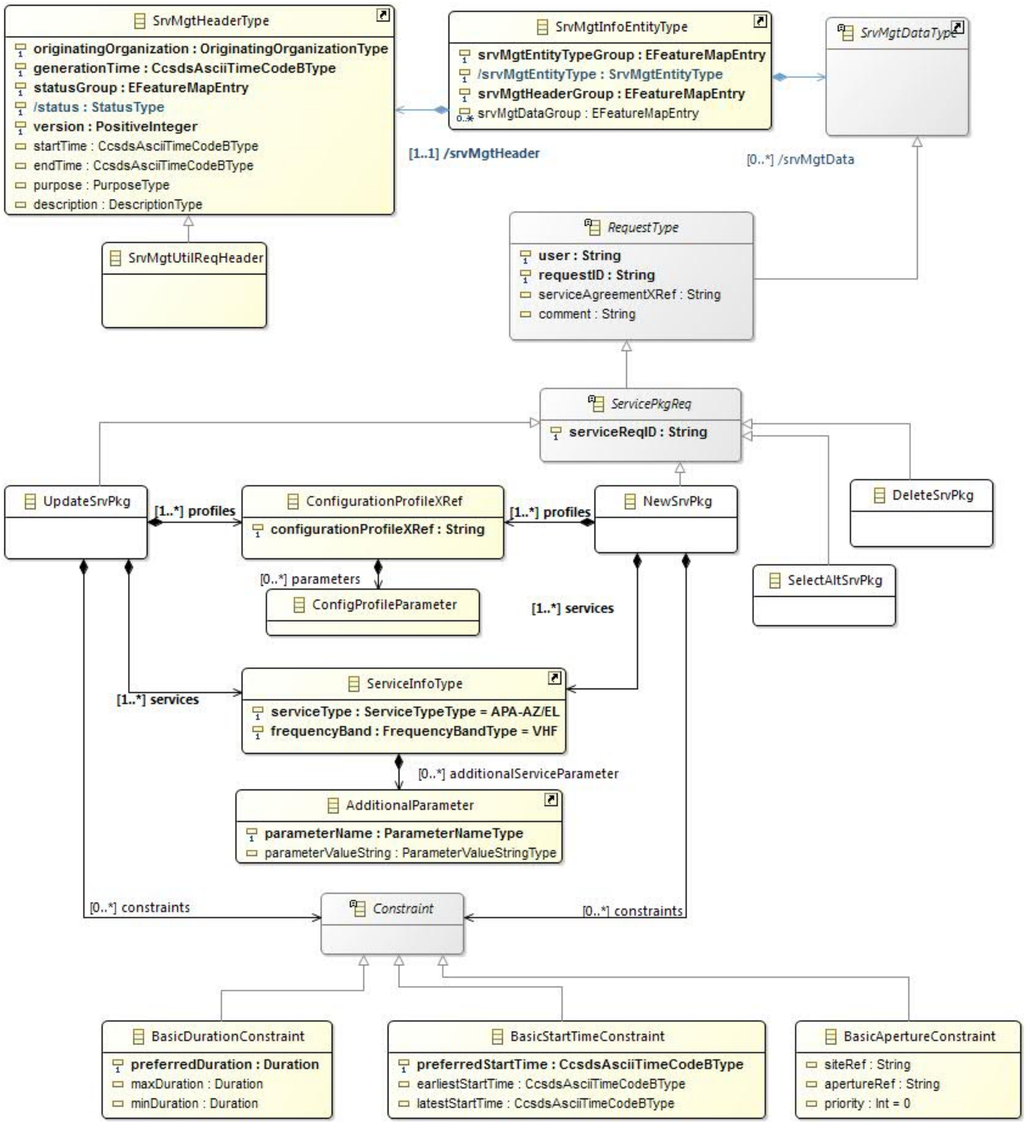

Figure IV-1 Utilization Request Formats

It should be noted that the above figure reflects what is in current draft of the SMURF. It has not yet been finalized and it is therefore likely that there will be changes before the standard is finalized.

Like the Simple Schedule above, the Utilization Request formats are based on common service management data types. To identify the request format, a **SrvMgtUtilReqHeader** extends the common service management header. An abstract **RequestType** providing common request properties serves as a basis of the still abstract **ServicePkgReq**. From here the four request types relevant in the current context are derived:

1. UpdateSrvPkg - *modify or update existing Service Packages*
2. SelectAltSrvPkg - *select alternative Service Packages*
3. NewSrvPkg - *request new Service Packages*
4. DeleteSrvPkg - *delete Service Packages*

It has to be noted that at the time of writing the service Package Requests do not have a direct correspondence to the elements provided by the Simple Schedule; the Simple Schedule as such does not contain Service Packages. It rather contains Scheduled Packages, which can in turn reference Service Packages. Consequently a mapping needs to be established if the Utilization Requests shall be used to requests elements which are reported back by the means of a Simple Schedule. We suggest that replies to Service Package Requests be considered as Scheduled Packages and mirror the **RequestType/requestID** as **SchedulePackageType/originatingRequestId**.
At the time of writing the Service Management Utilization Requests specification does not foresee any mechanism to query the state of an issued request, nor does it provide an acknowledgement mechanism to confirm reception of requests or state their rejection. In section V we will propose a mechanism how this can be mitigated if needed.
Note: The **ServiceInfoType** has been used is shown here as defined for the Simple Schedule, in particular the **ServiceInfoType/additionalServiceParameter** is not part of [2]. This is an anticipated harmonization for [2] with respect to [1].

V. What is not Covered

It has to be noted that the CCSDS recommendations for schedule interfaces do not contain elements to support negotiation among network users, report mechanism for conflicting requests or notifications for changes to implemented requests. For ESA these additional elements are part of the operational infrastructure [7]. Nevertheless it is assumed that for cross-support it is fair to start with a simpler interface to avoid complexity. If needed it is always possible to bilaterally agree on additional interface features for cross support.

VI. Transport Mechanism

As mentioned above the CCSDS Simple Schedule and the Utilization Request Format book specify data exchange formats. In order to actually establish a cross support interface for schedule purposes a transport mechanism needs to be put in place to exchange these data formats.

A. File Transfer

One option is clearly to perform a file transfer for the XML formats files specified by [1] and [2]. While any of the existing file transfer technologies could be used, these solution also require bilateral agreements until they are fully defined and usable in the context of cross support. For an interoperable file transfer CCSDS is working on a recommendation which , according to [5] is a 'CCSDS-recommended Terrestrial Generic File Transfer standard, which is designed to facilitate standardized exchanges of files and associated metadata between space agencies'. This standard is at the moment referred to as 'Terrestrial Generic File Transfer' (TGFT) and

- Mandates the use of a well specified transport mechanism
- Provides a file packaging mechanism for transfer purposes
- Standardizes an extendible meta data format to allow a unified and standardized processing to files according to meta data

To realize a CCSDS compliant schedule interface, [5] is certainly the candidate to consider.

B. Representational State Transfer - REST

Another common interfaces paradigm when it comes to XML based interfaces is the Representational State Transfer (REST); for a general introduction of the topic see [6]. REST is an architectural style and relies on well-established web standards like HTTP and HTTPS [8] for the transfer and management of data elements. In the current context the term data elements can be directly translated to XML documents, or more specifically to Simple Schedules and Utilization Requests. REST based interfaces typically use HTTP verbs GET, POST, PUT and DELETE to retrieve or delete elements or transfer them to a remote. In the context of a CCSDS compliant schedule interface REST can be used to transfer Utilization Requests to a remote Complex Manager and to query Simple Schedules. In principle REST allows the design of push or pull interfaces or any combination of them. It may be advantageous that REST relies on well-established web technologies, supports authentication (e.g. http basic authentication), encryption mechanisms (e.g. https) and provides good implementation support for several programming languages. In general REST has the reputation of being interoperable, which seems to match up well with CCSDS standards being about interoperability at the core.

VII. REST Schedule Interface for CCSDS Schedule Formats

In the following section we propose a basic REST interface for the exchange of CCSDS Utilization Requests and CCSDS Simple Schedules in an ESA implementation.

A. Authentication and Security

For the REST authentication HTTP basic authentication [9] over https shall be used[4]. The HTTPS / SSL certificate to be used has to be agreed between the Complex Manager and the Utilization Manager. In general the communication is secured by https.

Clearly the REST interface could also include account elements in the URI, but since it is any case needed to authenticate the user and restrict resource access for authenticated users, the account is not necessarily part of the URI.

B. MIME Type

All request and response bodies shall use MIME type mime text/xml.

C. Base URI

All REST methods described here are relative to a base URI, an example of a base URI is given below:

`https://example-network.int/schedule-manager/rest/1.0`

D. Return Codes

The HTTP return codes are defined below.

HTTP verb	Return
GET	200 – OK, 404 – Not Found
POST	201 – Created, 404 - Not Found, 409 – Conflict
PUT	200 – OK, 204 - No Content, 404 – Not Found
DELETE	200 – OK, 404 – Not Found

E. Get Schedule Query

The get schedule allows a Utilization Manager to query to get a CCSDS Simple Schedule.

Get Schedule Query

HTTP verb	HTTP request
GET	**/schedule**?start=2016-001T00:00:00Z&end=2016-010T00:00:00Z&filter=all_scheduled_packages&format=ccsds-r1

Parameter	Format	Description
start	UTC Date in YYYY-DDDTHH:MM:SSZ	The requested schedule start time
end	UTC Date in YYYY-DDDTHH:MM:SSZ	The requested schedule end time
filter	Allowed values: all_schedule_packages and values agreed otherwise.	The content filter
format	Allowed value is ccsds-r1.	CCSDS Simple Schedule according to [1]

Response Body

The get response shall carry an XML element compliant to SimpleScheduleType as specified in [1].

[4] Other methods like HMAC, OAUTh or OATH2 are available as well.

F. Utilization Request Submission
The utilization request od allows a Utilization Manager to submit a service package request to a Complex Manager. The attempt to submit the same request twice will overwrite previous requests.

Request

HTTP verb	HTTP request
PUT	**/utilization-request**/?requestID=4711

Request Body

One XML element derived from ServicePkgReq: UpdateSrvPkg, SelectAltSrvPkg, NewSrvPkg or DeleteSrvPkg.

Parameter	Format	Description
requestID	String, up to 1024 characters	The unique identifier of the request

G. Utilization Request Query
The resource schedule request method allows a Utilization Manager to query a request to schedule a resource(s) from a Complex Manager.

Request

HTTP verb	HTTP request
GET	**/utilization-request**?requestID=4711

Parameter	Format	Description
requestID	String, up to 1024 characters	The unique identifier of the request

Response Body

One XML element derived from ServicePkgReq: UpdateSrvPkg, SelectAltSrvPkg, NewSrvPkg or DeleteSrvPkg.

Note: we suggest to introduce a Request State for either the RequestType or ServicePkgReq to support tracking and explicit handling of request states.

VIII. Layering – Resource Configuration in the Context of Resource Scheduling, – CCSDS approach vs. ESA's common Practice

When scheduling a resource of a ground network, typically a ground station, it is clear that the resource must be specifically configured to fulfill the particular purpose. Both, the CCSDS Recommendations [1], [2], [3] and ESA's ESTRACK Management System (EMS) in charge of ground network scheduling, use the concept of Configuration Profiles for the purpose of configuring a ground resource.

While the CCSDS Service Management configuration concepts are similar to ESA's approach, there are differences. Here we focus on conceptual differences not on detailed syntactical differences.

1. While the CCSDS Utilization Requests use the Configuration Profiles at pass (Service Package) level, EMS handles Configuration Profiles at the level of individual services (e.g. Telemetry Service in a particular profile) provided within a pass.
2. The CCSDS Utilization Requests allow (but do not mandate) to submit a complete Configuration Profile along with the request to book (or update) a pass. In contrast to that ESA's EMS treats the Configuration Profiles of a ground network resources as something which can be referenced when requesting a ground network resource, but. EMS does not support the submission of complete Configuration Profiles.

Especially the latter difference seems to stem back from different approaches to configure ground stations for routine operations. For ESA's ESTRACK network all stations are configured and validated before for space missions can request the network resources or are scheduled based on standing orders. At pass time these pre-configured Configuration Profiles are simply activated via the ground station management system. At scheduling time the Configuration profiles are only referenced and can be refined per pass. Per pass refinement is based on so called 'pass through parameters[5]'. The whole complexity of Configuration Profiles (see [3]) is not exposed to the scheduling system EMS and protects the scheduling system to a larger extent from changes of Configuration Profiles. ESTRACK station configurations are created 'once per mission' as a preparative activity, so standardized configuration interfaces are not a top priority. Basically the configuration is handled in one layer and the scheduling in an higher layer. These two layer communicate by reference of Configuration Profiles[6].

Other networks seem to follow a different approach with a centralized configuration management. From this central configuration manager the configuration is 'pushed' to the ground network resources when needed, i.e. at the setup time of the ground station. For such an approach a standardized interface to submit configurations may be beneficial.

From an ESA point of view the aspect of a standardized configuration approach is less of a concern for actual routine operations, while it certainly has a role in preparative activities when ground network resources are agreed for a particular space mission. Especially in cross support scenarios that is perceived a major improvement. The main driver is to ensure that the configuration is valid to avoid failed support due to inconsistent combination of configuration values. Any support to ensure valid configurations is welcome, it may an advantage to handle that at the level of Configuration Profile definition.

[5] The name 'pass through parameter' comes from the fact that the scheduling system does not interpret parameters refined for an individual pass, but *passes it through* to the Configuration Profile active at the ground network resource.

[6] Similar differences exist for the handling of trajectory data. The CCSDS Service Management provide support for handling of trajectory data and allow Service Packages to reference the 'applicable' trajectory data for a Service Package. For ESTRACK the handling of trajectory data and the scheduling of passes is completely separated and handled a different level. Nevertheless it is ensured that at pass execution time the ground station is using correct trajectory data to support the pass.

IX. ESA's Adoption of CCSDS Utilization Requests and CCSDS Simple Schedule

Today the pass allocation planning and the production of the machine executable ground station schedules for the ESA Tracking Network (ESTRACK) is performed by the ESTRACK Management System (EMS). At the core of the EMS the ESTRACK Planning and Scheduling (EPS/ESS) is in charge of performing automated station allocation planning based on standing orders. The standing orders are organized in mission agreements and each mission participating in the ESTRACK allocation planning has exactly one active mission agreement with one or more standing orders active at a time. On top of the standing order based allocation planning missions can issue individual requests to create, modify or delete passes, which are in EMS terminology referred to as Operational Service Sessions (OSS). The OSS (modification) requests are called OSS Updates (OSSUPD).

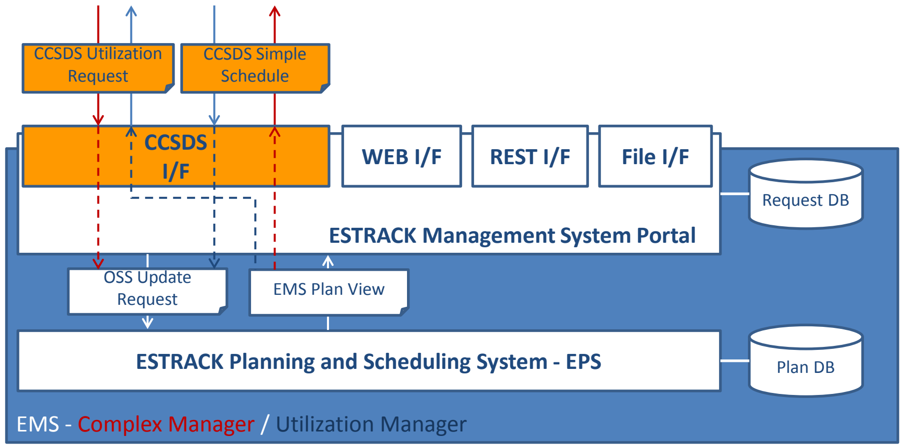

Figure VIII-1 ESTRACK Management System with CCSDS Interfaces

As indicated by Figure VIII-1, the ESTRACK Management System has an ESTRACK Management System Portal (EMS Portal) which allows ESTRACK users to interface the EMS via WEB interface, a REST interface or a File interface. Via these interfaces the EMS specific formats for requests and EMS Plan Views are exchanged. While the EMS OSSUPDs are the counterpart to a subset of the CCSDS Utilization requests, the EMS Plan View is a functional equivalent to the CCSDS Simple Schedule. To realize the CCSDS compliant schedule interface, the EMS Portal is extended by a mapping component to map the CCSDS Request and Schedule format to their EMS equivalents. In Figure VIII-1 the components in charge of CCSDS specific schedule formats are shown in orange and the mapping of data formats is indicated by dashed lines.

In its nominal mode the EMS acts as a complex manager performing the planning and scheduling of ESTRACK passes. For such cases the CCSDS interface is a pure format translation form the CCSDS Utilization Requests to the EMS OSSUPD format. In a similar way the EMS Plan View format is mapped to the CCSDS Simple Schedule format.

While the EMS is normally acting as a complex manager in charge of planning and scheduling the ESTRACK pass allocation, the EMS is also taking the role as a Utilization Manager when it comes to request and negotiation of external passes. This concept, also known as 'one stop shopping', allows ESA space missions books all passes via EMS, regardless of being provided by ESTRACK or an external network: All pass requests from a mission are submitted to EMS. Once the EMS has planned and scheduled the passes according to standing orders and requests, external passes are requested by EMS at the external network. At that point the EMS becomes a Utilization Manager on behalf of missions. Once the negotiation with the external complex manager is finished, the pass is flagged as 'Ext. Accepted', during the negotiation the state is flagged accordingly. ESA missions can follow the state of the negotiation via the available EMS portal interfaces and are notified about relevant state changes via the EMS Portal notification mechanism (email notification).

To provide an idea how the EMS Portal keeps track of request states, Figure VIII-2 shows the states OSSUPD requests. For external requests issued by the EMS Portal Figure VIII-2 shows the relevant states in orange. Passes explicitly requested by space missions or created by standing order based planning enter state 'Ext. not Requested', if they rely on ground stations of external networks. For networks with a CCSDS compliant schedule interface, as described in this paper, the external requests are formatted as CCSDS Service Package Requests, see section IV and [2].

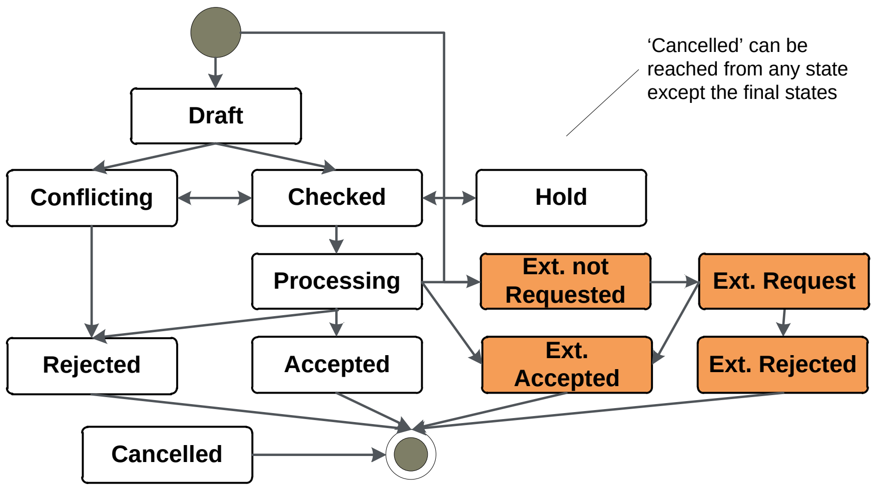

Figure VIII-2 EMS OSSUPD Request State Model

Once the requests to external network have been submitted, all corresponding requests transition to state 'Ext. Request'. In the case of CCSDS compliant schedule interfaces the CCSDS Simple Schedules are processed to assess the implementation status of submitted external requests. After mapping the Scheduled Activities into the internal EMS pass format OSS, the EMS Portal provides a so called 'Plan Diff' feature to support the EMS operator to incorporate the external schedules in a semi-automatic fashion into the EMS plan database: Confirmed passes enter state 'Ext. Accepted'. If the external network provider has rejected the pass request or has responded with a different result than requested, the EMS operator has the means to align the EMS plan database to the externally confirmed state or to delete the pass from the EMS Plan database.
To align the EMS plan database to the external provider, the EMS portal creates OSS update requests to either delete the pass from the EMS plan database (externally rejected) or to align it to the properties confirmed by the external provider. The EMS operator has also the option to wait, in order to address the (anticipated) situation that the external provider has not yet processed the request.
Last but not least the EMS supports updates of external requests. In this case the request (OSSUPD) is first processed internal to ensure consistency. Then the external request is submitted again to the external provider like described above. In principle also the cancellation of an external request would be support by EMS, however it is not clear how this can achieved with the current CCSDS Utilization Requests [2].

X. Conclusions

In this paper we have introduced the CCSDS recommendations relevant for schedule interfaces. A the moment these recommendations are in draft state and subject to change, however we have shown how these draft standards can be implemented at the example of ESA's adoption approach. A crucial property is the extensibility of the CCSDS schedule standards. Without such an extensibility an adoption would not be possible without changing the standard, although the use of extensions comes at the cost of the extension definition itself.

The CCSDS recommendations have been put into the context of CCSDS Service Management [3] and it has been stressed that implementation of the CCSDS recommendations relevant to ground network scheduling can be performed without having CCSDS service management in place, while allowing a later adoption. We believe that this is an important practical consideration. Finally the paper has shown that although space agencies may have different concepts in place to address the configuration of network resources in the context of resource scheduling, CCSDS provides mechanisms to support different concepts.

References

[1] CCSDS Simple Schedule Format Specification, Draft Recommended Standard, CCSDS 902.1-R-1.15a

[2] Service Management Utilization Request Formats, Draft Recommended Standard, CCSDS 902.9-W-0.01

[3] CCSDS Space Communication Cross Support—Service Management—Service Specification. Blue Book. Issue 1. August 2009.

[4] ESTRACK Support For CCSDS Space Communication Cross Support Service Management, H. Dreihahn, M. Unal, A. Hoffmann, DASIA Conference 2011

[5] Terrestrial Generic File Transfer, Draft Recommended Standard, CCSDS 927.1-W-0.00, November 2015

[6] Representational State Transfer, https://en.wikipedia.org/wiki/Representational_state_transfer

[7] ESTRACK Management System Portal - A Web-based Collaboration Platform for Ground Stations Planning, Alexander Hoffmann, Holger Becker, Tiago Simoes, Matteo Renesto, Holger Dreihahn, Martin Unal, Spaceops 2014

[8] HTTPS, https://en.wikipedia.org/wiki/HTTPS#References

[9] Basic Authentication, https://en.wikipedia.org/wiki/Basic_access_authentication