**Service Agreement and Configuration Profile Blue Book**

**Concept Description**

NOTE: Pietras comments are contained in [JP - square brackets in red].

Use of the configuration profile approach is the de facto method that TT&C networks employ for configuring space link sessions (a.k.a. contacts, passes, tracks, Events). However, the term “configuration profile” is itself the CCSDS name for this entity – actual Provider TT&C networks currently have their own network-dependent names (e.g., SN service specification codes (SSCs), NEN Support Activity Codes).

The service agreement does not have similar counterpart in the real life, as to now. The missions and network providers agree many aspects of the support in form of custom formatted documents or contracts. This lays the ground for further operational support, however is a tedious task and does not allow for real automation.

The Space Communication Cross Support Service Management (SCCS-SM) Service Specification, published in 2009, included standard XML-formatted Configuration Profile and Service Agreement specifications. The Silver-1 Configuration Profile was designed as part of a Configuration Profile Service, in which new configuration profiles could be generated dynamically by a Mission and submitted to the Provider CSSS shortly (minutes to hours) before the submission of a Service Package Request that referenced that Configuration Profile. The need for a Provider CSSS to be able to quickly validate a Configuration Profile led to a highly-formalized Service Agreement standard against which the individual parameter values in the dynamically-generated Configuration Profiles could be automatically and quickly validated. This, and few other shortcomings (monolithic structure, rudimentary timing offsets, all CCSDS or nothing nature) inhibited the implementation. Therefore, in the frame of new approach to the Extensible Service Management, the new structure of Configuration Profiles has been decided.

The recommended standard will be composed of a Service Agreement and Configuration Profile definitions, moving the service agreement away from previous setting together with Service Catalogue. Service Agreement wins with that more on actual technical importance for the complete lifecycle, and is recognized as integral part of the mission establishment and execution process.

Generally the Blue Book will follow concepts as defined in tech note *Requirements for Simple Configuration Profiles and Service Agreements*. This tech note is required for the deeper understanding of all dependencies and especially the coupling with the Functional Resource world, as the Blue Book won’t go into so much detail.

[JP - If the Tech Note is required to understand the Blue Book, then it (or the needed parts of it) should be turned into a Green Book so that it can be referenced by the Blue Book and available to any user of the Blue Book (i.e., be available via www.CCSDS.org.]

The Blue Book is a one consolidated recommended standard for two actual information entities: the Service Agreement and the Configuration Profile. They will actually be treated in separate sections, showing however the dependencies between these two, as required.

The Service Agreement chapter will include, beside the main definition of the data format (with UML diagram), the sections related to the Configuration Profiles themselves (or actually their containment), as well as so called Persistent Information.

The Configuration Profiles (itself defined in detail in Configuration Profile chapter) are solely either referenced or contained in a scope of Service Agreement. In other words, all Configuration Profiles will be created as part of the Service Agreements themselves, eliminating requirements for dynamic creation of configuration profiles and therefore eliminating the need to dynamically validate configuration profiles. The Configuration Profiles will each contain the ranges and sets of allowed values that their configuration parameters can take on, these allowing specified parameters in SMURF Service Package Requests to be validated against information that is contained in the Configuration Profiles themselves.

Another advantage is that when requesting a space link session be scheduled, instead of asking for an explicitly-detailed set of resources and associated configuration parameter values, the Mission simply references one or more configuration profiles that have been previously negotiated between the Mission and the Provider CSSS, and have been defined within the Service Agreement.

In summary, this section will focus on showing the dependency between actual Configuration Profiles and the Service Agreement, and how the containment shall be performed and understood, also in terms of lifecycle. Also, the considerations related to parameter value ranges or lists may take (some) place here.

The Persistent Information shall be defined as relatively simple, in terms of XML structure, set of classes containing specific information related to the complete Service Agreement. This would include the list of bilaterally agreed apertures, data storage policies or general booking constraints (like maximum number of scheduled Service Packages in a time period).

The Configuration Profile section will focus on three aspects: general data format definition (and its dependencies with respect to Service Agreement), the pre-defined service profiles for most common services and finally some considerations for user definitions of configuration profiles.

The general description of Configuration Profile contains description of the main classes and the dependencies of their combinations in form of Functional Resource Groups\* (like *Aperture*, *TM Coding and Modulation*, etc…). Additionally, for the use at the level of Service Agreement, the possibility to define parameter value ranges or lists is described. This section shall allow for basic understanding and implementation of the CSS Service Management compliant Configuration Profile (as a part of Service Agreement or an entity being referenced out of Service Package) without actually tackling on any deeper Functional Resource aspects (like individual parameters, etc…).

[\*JP – This may seem like a nit-pick, but terminology here is important. We should **not** call these “Functional Resource Groups”: that is a term I tried to use (now) years ago in the development of FR concepts and it was confused with the SLE concept of “Functional Groups” (see CCSDS 910.3-G-3). Over the intervening years I tried other sets of terminology that for various reasons had their own problems, until I proposed the *Functional Resource Strata*/*Functional Resource Set* terminology, which seems to not clash with anyone’s notions of what they should be. The FR Strata are the abstract layers into which the technology-specific FR Sets fall. The examples provided parenthetically are a mix of these concepts: Aperture is an FR Stratum in which multiple aperture-technology-specific FR Sets can reside (the only one that exists now is the RF Aperture FR Set, but we anticipate one or more optical-technology FR Sets in the future). The second example (TM Coding and Modulation) is presumably an FR Set because it is specific to the “TM” coding “technology”. However, the example itself is incorrect because it straddles two FR Strata: the Physical Channel stratum and the Sync and Channel Coding stratum. For the purposes of the sentence in question, the topic is FR Sets, and so the sentence should read something like “The general description of Configuration Profile contains description of the main classes and the dependencies of their combinations in form of Functional Resource Sets (like *RF Aperture*, *CCSDS 401 Return Physical Channel Reception*, *TM Sync and Channel Decoding*, etc.).”]

Second section shall cover for so called cookie-cutters. The actual detailed cookie-cutters in form of XML schemas, including pre-defined lists of CCSDS Functional Resources required to be configured for specific service execution will be stored at SANA registry, and thus being available for everybody for quick check against it. This way, the Blue Book will not focus at all on definition of functional resource parameters, (redundant to the actual FR definition also located in SANA registry), but will only mention them as required for fully CCSDS conform usage and focus only on Functional Resource ~~groups~~ sets and their required combinations for each specific Service Profile type. The cookie-cutters defined in the Blue Book will therefore walk through number of identified services, which will be explained in some detail and especially their definition in terms of combination of the Functional Resources required configuring specific service.

[JP – To my understanding, the “cookie cutters” are pre-defined configuration profile schemas that a Mission can take “off the shelf”, assuming that the Mission can use that specific standard profile. The cookie cutter config profiles are created from pre-defined Space Link Service Profiles, which in turn are defined from the FRs in FR Sets. I think of 2 levels of aggregation at which the XML schemas can be maintained in SANA. At the lower level (call it level 1), the schemas for just the FR Sets would be registered in SANA. Level 1 would require the Blue Book to (a) specify each of the cookie cutter config profiles and (b) identify which SANA-registered FR Set schemas to use to build each of the config profiles. At the higher level (2), SANA would contain not only the schemas for the individual FR Sets but also the aggregate schemas for the cookie-cutter Space Link Service Profiles (SLSPs) and configuration profiles for those SLSPs (probably as schemas that simply include or import the appropriate FR Set schemas). Level 2 would still require the Blue Book to specify each of the cookie cutter config profiles, but could just point to the single configuration profile schema in SAN for each of the cookie cutter profiles. The individual FR Set and SLSP schemas would be available for use by those missions that need to create configuration profiles that are satisfied by the cookie cutters. I propose that the Level 2 approach be adopted.]

The actual content of each Functional Resource – their detailed parameters – are treated in two ways. First the general definition of freely defined parameters or parameter sets (which may be bilaterally agreed between parties) and secondly the reference to the respective fully CCSDS conform (in terms of Functional Resource definition) XML schema located at the SANA registry.

That way the implementing organizations may choose if they to freely decide on contents of the Configuration Profile, still conforming in total to the Functional Resource dependencies within specific Service or completely support the existing set of Functional Resource parameters already existing at SANA. The way of treating that this way in Blue Book, is a tradeoff between keeping the book development in reasonable time and on the other hand do not completely loose the Functional Resource concept out of sight. This consideration is also of an importance with respect to the extent of the book, which already just covering simple FR combinations for selected Services is going to be very extensive. Additional definition of each single parameter would inevitably lead to huge incomprehensive recommended standard. Another issue is in that case the synchronization between actual SANA registry of FR and their definition in the SACP book. [JP – yes, a significant issue. ]

As already mentioned before, in order to validate specification and reconfiguration, the allowed range/set of parameter values for each parameter may be included as part of the Configuration Profile itself. This gives in general the Service Agreement boundaries during mission establishment process, and later on allows in elegant way for each Configuration Profile to be self-verifiable.

Finally the book will discuss some topics related to multi-service Configuration Profiles and their interactions.

**Summary**

SACP Book will:

* Include definitions for two CSS SM information entities: Service Agreement and Configuration Profile
* Follow concepts of the
* Service Agreement will be essentially a collection of Configuration Profiles augmented with allowed value ranges or lists
* Service Agreement will contain also so called Persistent Information
* Configuration Profile format definition (and thus the main schema) is very high level, thus theoretically allowing free profile definition depending on mission needs (also outside of Functional Resource universe)
* Configuration Profile will be supported by so called cookie-cutters, representing pre-prepared combinations of Functional Resources, which follow the FR concepts for typical services (i.e. Forward CLTU Space Link Service).
* The exact parameters for each Functional Resource within each selected Service may be either freely defined (bilaterally agreed) or may follow strict definition with actual Functional Resource parameters assigned to each Functional Resource. This selection can be done on mission to mission basis, and the respective SANA stored schemas will contain current set of CCSDS-configurable parameters, which may be used.

[JP – At one time Marcin was proposing a generic FR schema that could be substituted into a configuration profile in place of one or more CCSDS-standard FRs. I don’t know if that idea is still part of the concept. This approach might work if only scheduling were the concern, but I think that it would fall apart when monitoring and real-time control are brought into consideration. The better approach – and one that conforms with the FR model – would be for each Agency/network to register their Agency/network-unique FRs on the SANA registry subtree that is already set aside for that purpose. That will establish the FR Type (OID) that is necessary for parameter values and directives to be named and thus exchanged via MD-CSTS and SC-CSTS. Note that these Agency/network OIDs themselves don’t have to be registered in SANA unless they are expected to be used in cross-support.]