SCCS SM Event Sequence

Fall Meeting, November 11, 2015

Paul Pechkam, JPL/NASA

Status

- Spring 2015 – Pasadena
  - Agreed to look at Functional Resource Model (FRM) integration
  - 1st look at Service Control with joint mtg CSTS and SM working groups
- October 12, 2015 Service Control and ES splinter group telecon
  - Different perspectives of CSTS-SC and SM-ES presented/discussed
  - Agreed to do a closer look at FRM integration
- Today
  - Revised state model that integrates FRM states
  - Several approaches for integrating FRM in ES information entity
  - Ongoing development/analysis
    - Refactoring to be consistent with revised configuration profile
    - Analysis of service control execution impact to event sequence execution
    - Revised/additional use cases - Additional flexibility in describing “relative” events/states and their sequencing
  - Book sections updated to explicitly refer to FRM
    - Parameter tables updated to FRM types

Event sequence – background

- The event sequence is an input to refine the scheduling of provision between the user and provider within the scheduled times of a scheduled service package
- Its goals are to allow a user to specify sequence of events without having to maintain or manage knowledge of provider internal resource workings, behavior and process
  - Spacecraft spacelink – as a means for users to specify
    - Specify characteristics of the spacelink for ground communications to lock to a return carrier and/or uplink to the s/c and establish coherent forward and return carriers without specifying/configuring specific receivers, xmitters, etc
  - Scoped to spacelink session sequencing
- Its main concept is (still?) to allow the user to specify the requested space links and data transports as a function of time using a configuration profile(s) and/or inline state parameters to specify the differences in communication state

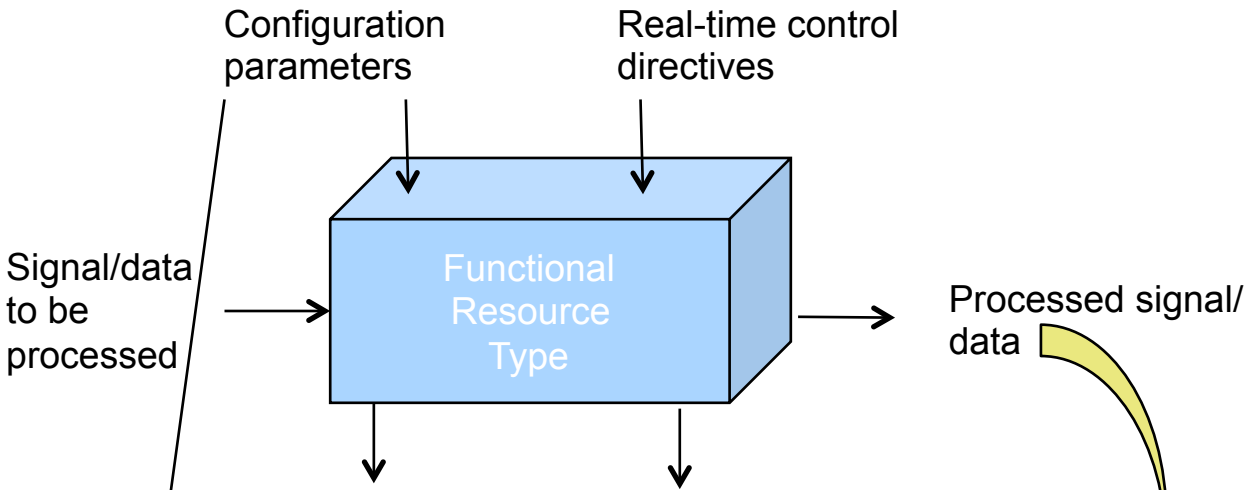
Event sequence – changes

- What is different?
  - With FRM, those space link characteristics are now controlled through FRM configuration parameters
  - Configuration specification is now using ESLT “ground” terms
  - Inclusion of the FRM sounds like a good idea since it provides a common reference framework with configuration profile and CSTS-Service Control. In theory, directives could be traced to FR instances and hence impacted states

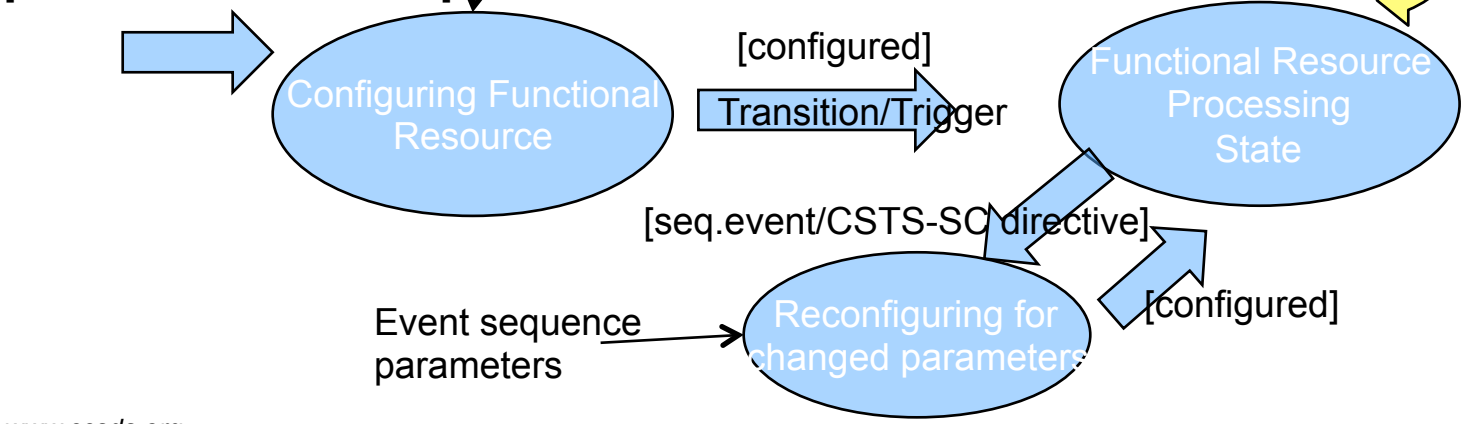
USING THE FUNCTIONAL RESOURCE MODEL

Analysis - FRM resource type state modeling

FRM component model (from tech note)



Derived FRM states

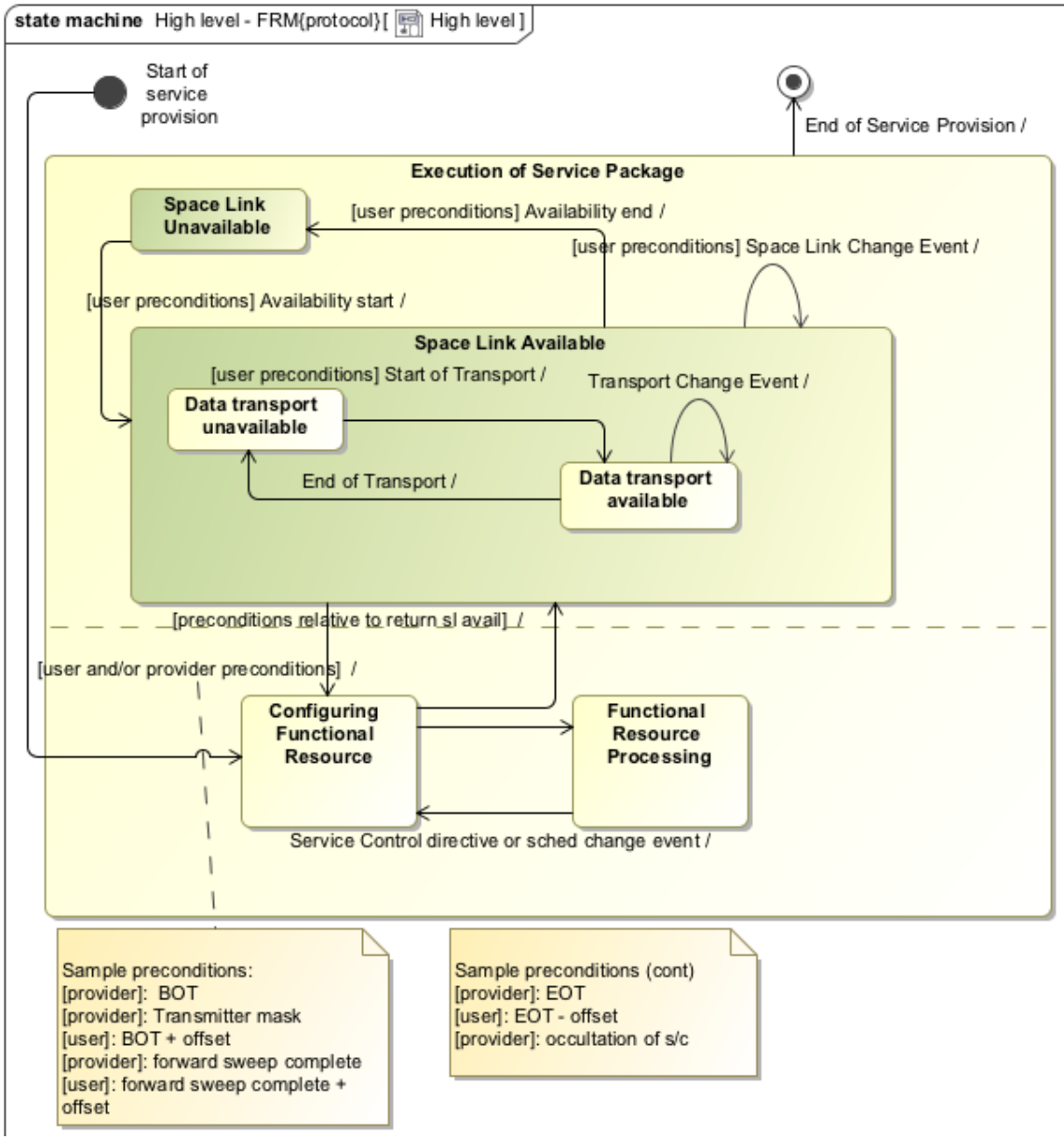


In this state, the processing signal/data is present on the SAP

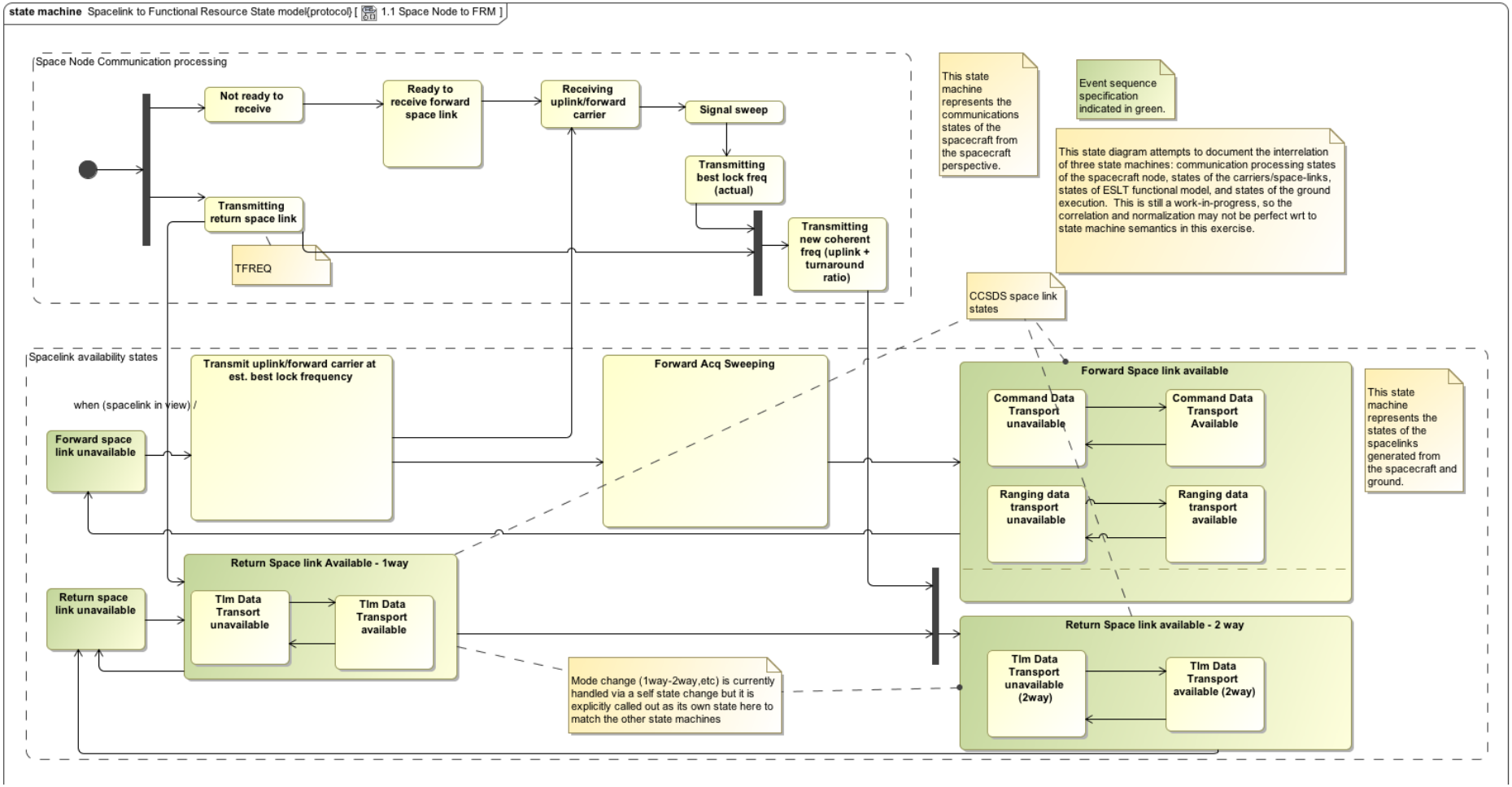
Revised service package state model

1. FRM configuring/processing states added to state model – parallel to spacelink states
2. Preconditions/transitions to allow for user to specify sequencing using state relative terms ie starting transmitter earlier or waiting until 2 way/coherent space link (ie after sweep and lock) before starting data transports
3. Service control directive appears (so far) in the FRM states

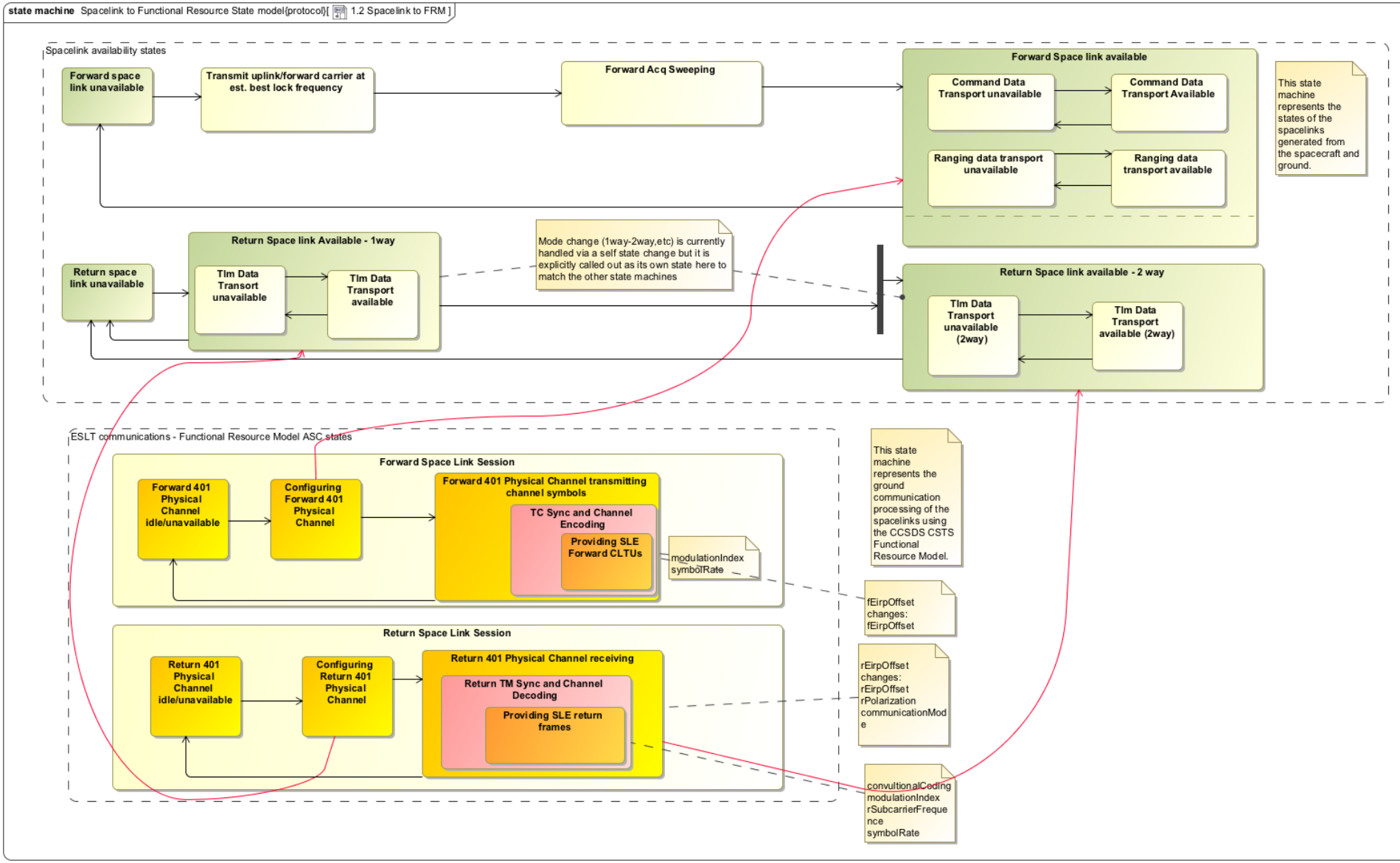
The following two slides document more detailed state models for your review or background info



Detailed state model



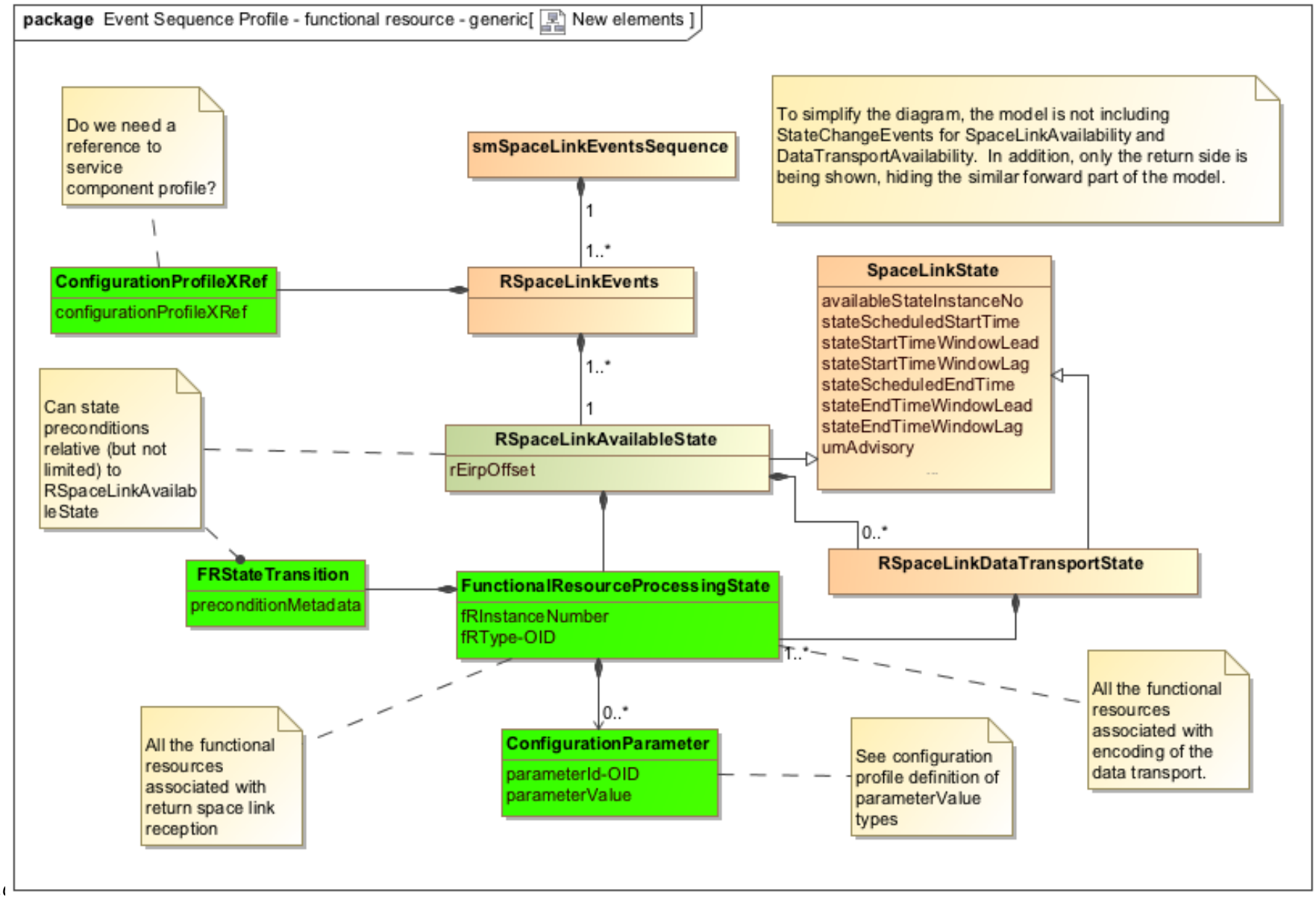
Now with FRM (and a new ground/ESLT perspective)



State modeling analysis

- Inclusion of FRM is the combination of the spacelink availability state machine and the FRM ESLT state machine
  - We now explicitly refer to ground configuration and changes
- Information entity refactoring - Not much different structure in the ES, however
  - The relationship between space link carrier and its parameters are more explicitly defined with respect to functional resources classes and expressed in functional resource terms/context
- How to combine the FRM structure and the ES structure
  - Well, there seem to be a couple ways…
    - Do we normalize to states or reference FR types
    - Do we define each parameter, or
    - Do we use generic parameter a la configuration profile approach (schema is “unaware” of specific configuration parameter types)

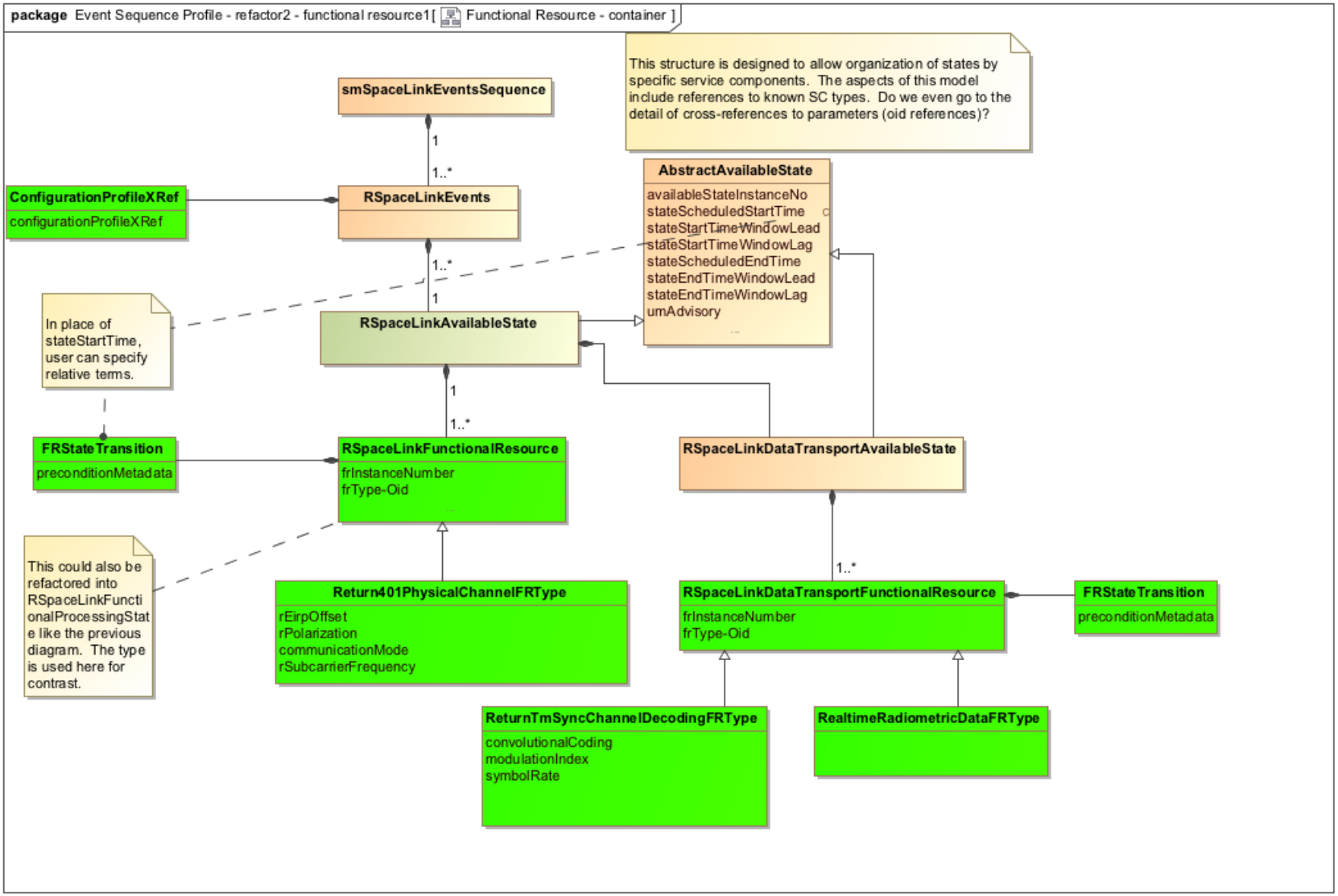
Approach #1 – Generic parameter re-specification



Approach #1 – Generic parameter re-specification

- Philosophy different from current approach – generic parameter specification is broad approach to configuration
- Post-schema syntactic validation
- Semantic validation needed against Service component and combination profile(s)?
  - Implies some governance from the configuration profile on ES e.g. do we need to scope what can be changed in a particular state or change event?
- Extensible – no changes needed for new or revised functional resources

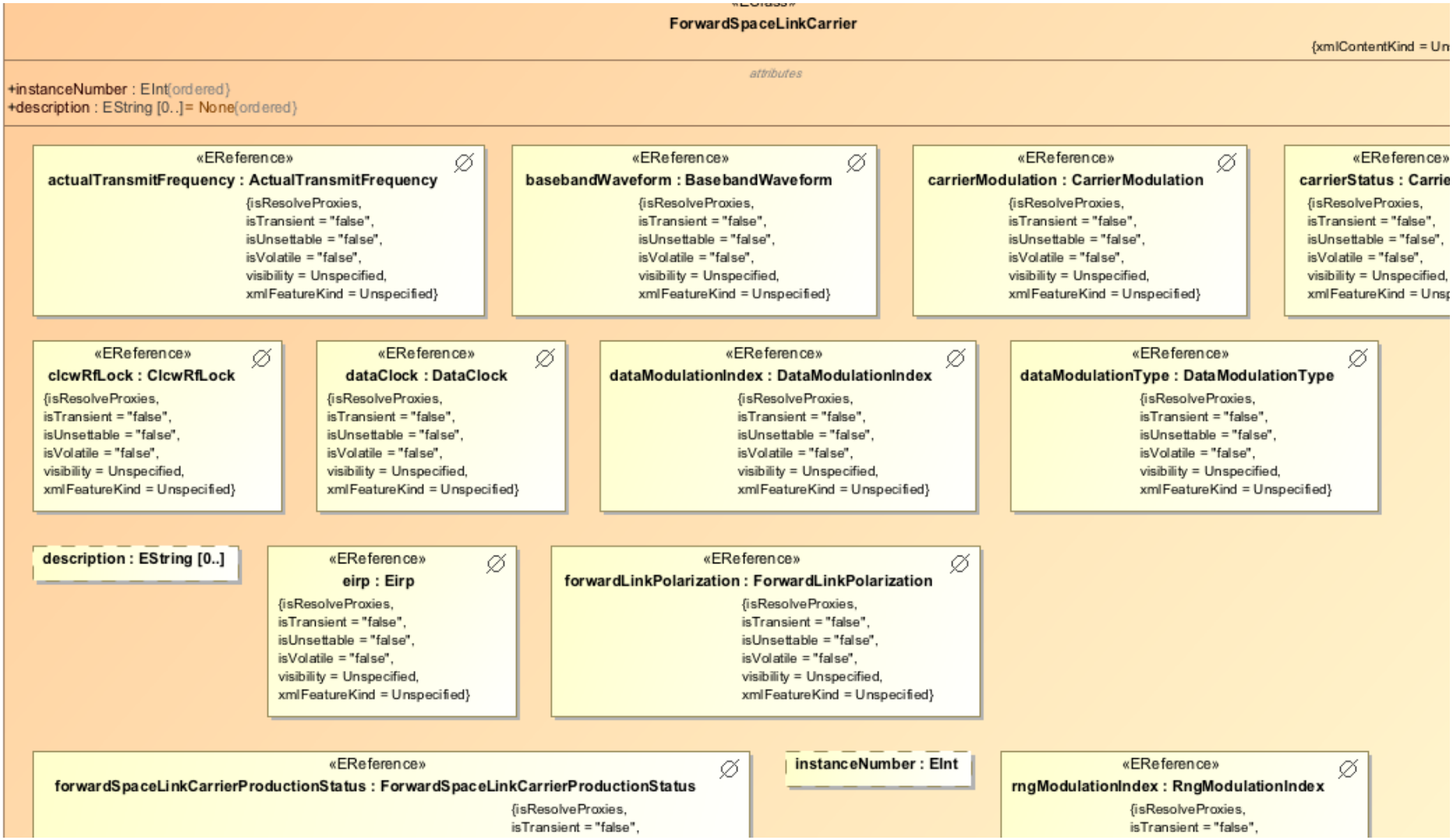
Approach #2 – parameters, FR types defined



Approach #2 – parameters, FR types defined

- FR Types are explicitly defined (similar to blue 1 approach)
  - Should they be FR Type “processing” states?
  - Or FR Types (component) to maintain consistency with configuration profile and (see next bullet)
- Possibly leverage FRM model instance transformation to keep SM ES consistent with latest FRM? (see next slide for example)
  - One idea is to use generated classes where FR types/states are used
  - Implies event sequence constraints or transformation rules on FRM
  - Holger performed transformation exercise from FRM to MagicDraw and eclipse import files
    - No configuration parameters (yet) – only monitoring attributes
    - But do we need to constrain to parameters only used in event sequence – is that easy/worthwhile to do?
- Would need extension points to add additional parameters

Transformation from FRM



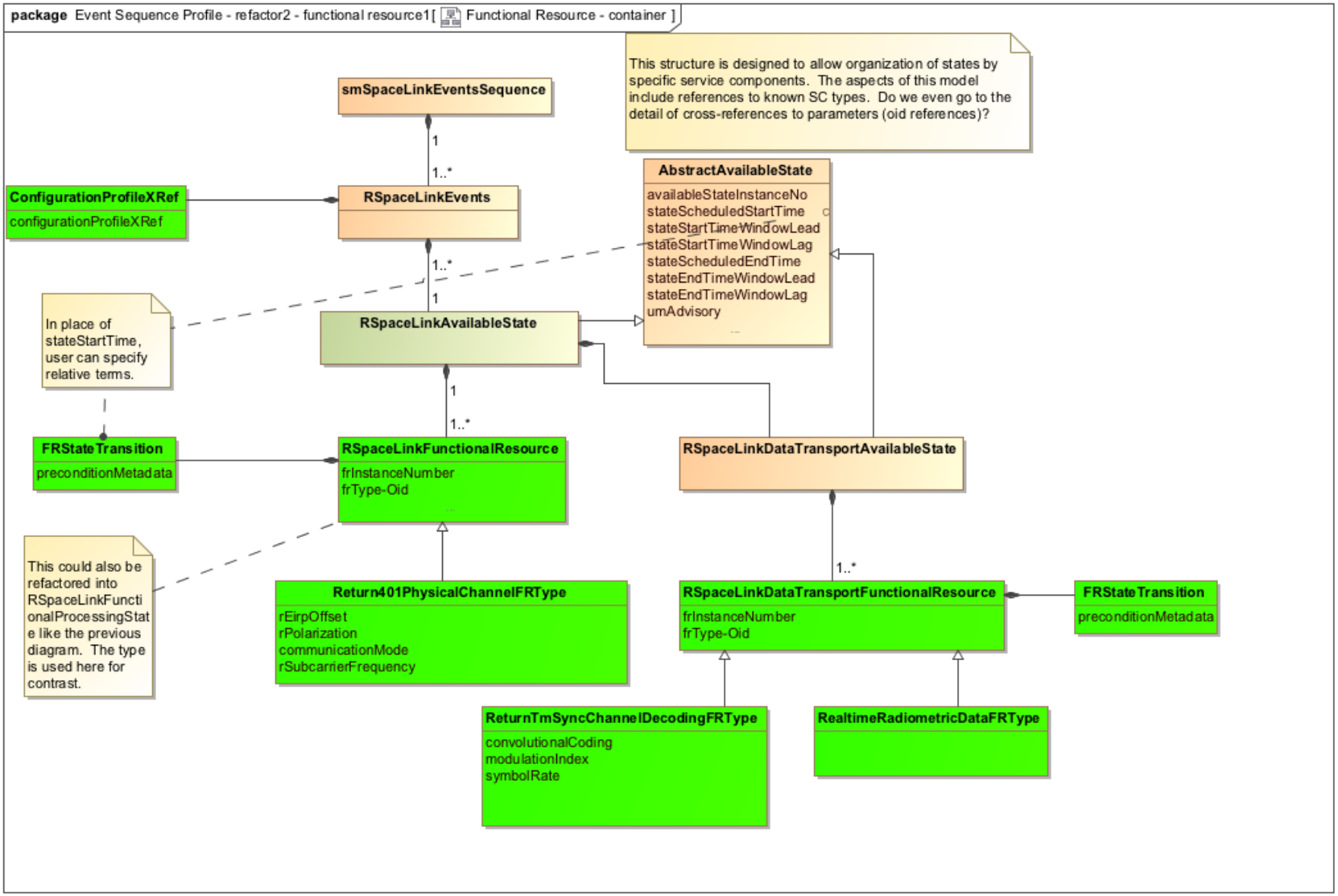
Service control

- Transitioning to FRM is a step towards being able to
  - Evaluate and determine impacted states
    - Better – be able to directly reference impacted states?
  - Write rules for how to handle specific directives e.g if directive to change forward link EIRP, override EIRP in current forward link availability state
    - Future forward link states would use the event sequence value
    - Do these appear as part of management services?
- Use cases
  - Changing a configuration parameter
    - Since ES is now using the FRM, a Provider should be able to correlate an SC directive (using FRM parameter reference) to an ongoing state and executing FRM instance
      - Can now make decisions on how to manage ES execution (current and future states)

Offsets, indefinite states, etc

- Today's operations reveal preference/offsets when establishing spacelink or data transport availability states (see also Erik's state diagram)
  - Stating SL availability start time, state SL availability relative to Provider Beginning of track (BOT)
  - 2 way Return SL availability relative to BOT + offset + forward sweep + RTLT
  - Wait for 2-way coherent forward/return SL's + 30 sec before starting command data transport
  - Etc
- Need to be able to express offsets and/or relative terms
  - Does the Configuration Profile FlexibilitiesAndConstraints achieve (all of) this?
  - Rules may be specified generically using a metadata field (like in simple schedule association) Or provide a standard set of rules and additional reference framework (for relating to states)?

Approach #2 – parameters, FR types defined



ADDITIONAL BACKGROUND/ BACKUP MATERIAL

Further service control analysis

- Isn't what the provider shall provide as a 'sequence of events' the set of CCSDS (CSTS) services and
  - when they are provided
  - in which configuration, i.e. parameterization?
- Answer – the Blue 1 approach is that it is a sequence of spacecraft tracking events that state the characteristics of the space-link(s) between the space node and earth nodes
  - Only the data services related to telemetry, commanding and ranging (that can be characterized through space link characteristics) appear here, excluding the data transfer services (which directly handle the data between provider and user)
- Blue 1's goal was to remove the user from having to specify parameters for specific (to the provider) ground resources – ie finding the right abstraction for standardization

- FRM is intended to standardize resources – allowing it to be used in event sequence