

Interface Patterns Project

Meeting Minutes

Date/Time: Friday, May 5, 9:00 – 11:00 AM EST

Attendees:

Bill Schindel	ICTT System Sciences	schindel@ictt.com	812.232.2062
Frank Salvatore	Engility Corp.	frank.salvatore@engilitycorp.com	973.607.2068
Jon Torok	NSWC Crane	torokrj@gmail.com	812.854.5247
Jason Sherey	ICTT System Sciences	sherey@ictt.com	812.232.5968
Stephen Lewis	ICTT System Sciences	lewis@ictt.com	812.231.7930

Summary:

1. We made progress on understanding some key questions about the most abstract components of the interface patterns—the S*Metamodel and near specializations thereof, and have identified a few related model questions to get resolved.
2. We agreed how to form our priority interface pattern targets for discussion in the next meeting.
3. Some questions about the limits of what Magic Draw supports in the way of profile-based aids to users were answered for the current release.
4. Magic Draw, Inc., has agreed to provide a form of cloud-based model support facilities in support of the project.
5. We agreed on action items, and the next meeting will be on Friday, May 19, 9:00 AM EST.

Details:

6. Magic Draw, Inc., Cloud Service
 - a. We don't know yet whether this will be Team Work Cloud or Team Work Server
7. MD/CS Profile:
 - a. Jon answered Stephen's question about showing relationships graphically in the profile: While this improves the profile as documentation, apparently MD does not provide direct profile-enforced relationship constraints or aids when constructing subsequent models governed by the profile. (Some SysML tools do to a degree, some don't.)
 - b. In any case, we will be interested in the related similar case for parent (superclass) S*Patterns and their child (subclass) S*Patterns, at each level of the class hierarchy. (These are the S*Gestalt Rules.) So, at least a query to show inconsistencies will be of interest later, and serves a similar purpose to the currently missing MD/CS profile-level capability. Such a query aid will also help users of Interface S*Patterns.
8. MD/CS Attribute Value Tables:
 - a. The attribute value tables we had used can leverage table view capabilities of MD/CS, according to Jon.
 - b. A key use we make of these in S*Patterns is for configured patterns, which will be configured S*Interface Patterns in this project, as might be used for ICD document tables, etc. For follow up with Jason or Stephen.

9. S*Metamodel Level Interface Discussion Questions:

- a. Bill recalled the S*Metamodel level definition of Interface (see S*Glossary distributed):

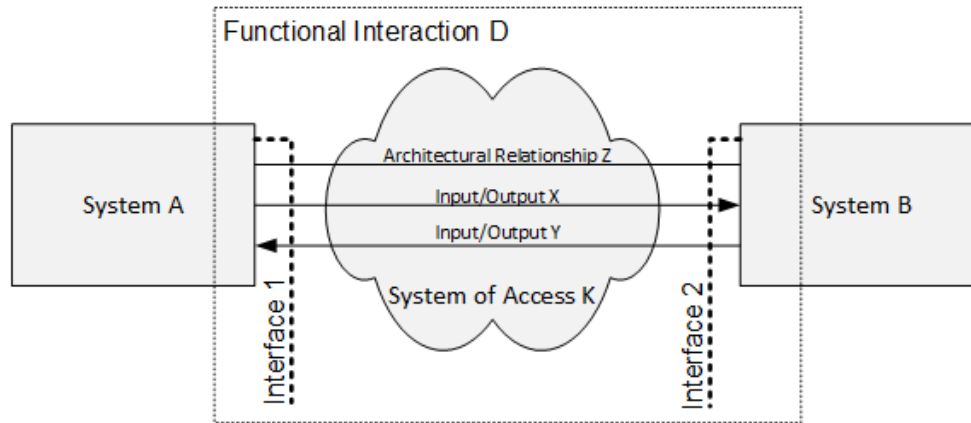


Figure 1: Diagram supporting definition of Interface

In S*Metamodel, an Interface is an association of:

- A System, which has the Interface;
 - One or more Input-Outputs, which pass through the Interface;
 - One or more Interactions, which describe the behavior at the Interface;
 - A System of Access (SOA), which provides the means of interaction of the interacting systems.
- b. Jon then asked whether the combination of the SOA and surrounding Interfaces (see above diagram) might be called something specific.
- c. Jason pointed out that it was close to another System of Access, and Bill noted that it would be an SOA if combined with some small roles that were internal to System A and System B of the above diagram. Refer to Figure 2 below, for a sequence of diagrams that all reflect different views of the same situation.
- d. Figure 2(a) represents an interaction exchange of Input-Output X between two systems, same as Figure 1.
- e. Figure 2(b) looks inside the interacting black boxes of Figure 2(a), revealing internal communication stack roles A1, A2, A3, B1, B2, B3 (e.g., electrical line drivers, communication stack subroutines, mechanical points of attachment, etc).
- f. Figure 2(c) shows that the combination of the original SOA K with Interface 1 and 2, along with the additional internal SOAs, together forms a larger SOA L.
- g. Figure 2(d) shows that this same situation can be viewed as a larger SOA with interfaces (now internal to the two original systems A and B) at the “application level”.
- h. This sequence illustrates the underlying difference, highlighted in the S*Metamodel, between Interface and System.
- i. For additional discussion in our next meeting.

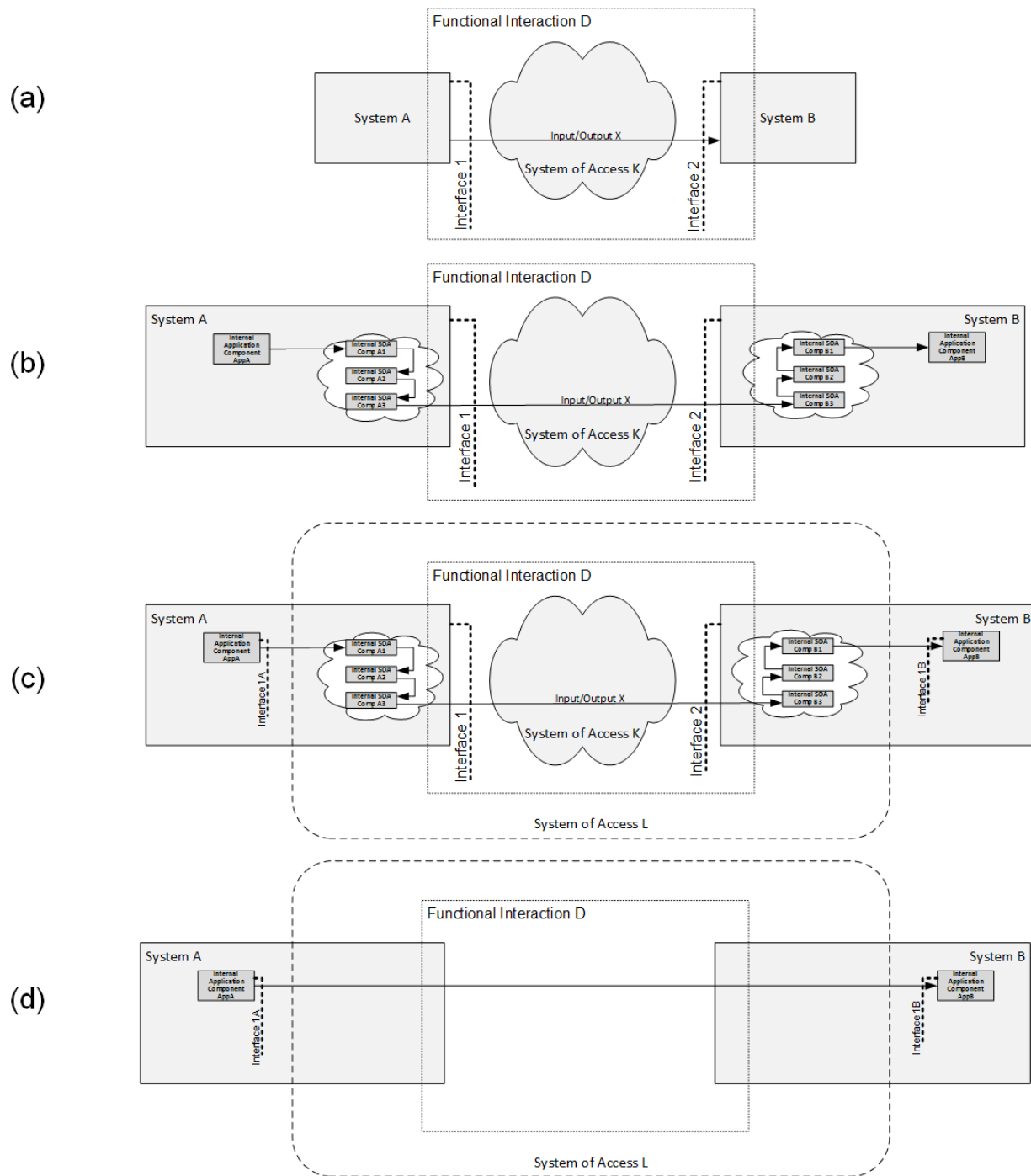


Figure 2: When enlarging SOA—sequence above—other SOAs can be seen, as in Communication Stack Interface, or similar

10. Independence of Individual Tools; Compliance with SysML; Use in Non-modeling Tools:

- a. Jon noted interest (we all agreed) in maintaining SysML compatibility of the Interface Patterns, encouraging others to be able to use the resulting patterns. Frank noted interest (we all agreed) in the same underlying Interface Patterns also being usable in other tools and non-modeling databases. Jon noted that his organization would also like

to have a DB that is not SysML, but can interoperate with SysML. Bill noted that those are direct purposes of the intermediate layer S*Metamodel, which is mapped to SysML for that toolset, but also mapped to DOORS, PLM systems, and other databases for use of the same portable patterns in other toolsets. A key feature of the Interface Patterns will be their portability across toolsets and databases, which we should plan to demonstrate. This points out that the real significance of the earlier discussion above is the underlying meaning, independent of toolset, of the concepts of Interface, SOA, etc.

- b. Bill noted he is giving an invited talk at the No Magic MBSE Symposium this month on maturity of MBSE methods and tools; Jon noted that he may also be there.

11. Work Toward List of Targeted Interface Types for Patterns:

- a. Bill showed a draft slide of “possible work streams”, which was edited during team discussion resulting in:

Possible work streams

1. Identify interface aspects of the S*Metamodel (the most abstract interface pattern)
 - List of those aspects
2. Create library of interface patterns of different types (specializations of 1) showing techniques in mechanical, communication, visual, etc.
 - List target interface classes (Jon suggested mounting, communication, ...)
 - And how to use them
3. Identify queries and views that are interface-based (e.g., ICD, etc.)
 - List the views and queries
 - What metadata should appear in each of these (what metadata in an ICD, etc.)(converting these to S*Metamodel terms is a good way to learn it as well as test it)
4. Identify interface-oriented tasks, activities in the engineering life cycle (the reasons we are doing this stuff)
 - List them
5. Down the road, issues of governance of the resulting patterns, their life cycles
6. Tactical level tool specific items, not necessarily interface-oriented, along with mappings to SysML or specific tools
 - List

Figure 3: Work stream list edited during meeting

- b. It was agreed that each individual would generate a list of highest priority interface types we would like to see in work plan, and exchange by email before next meeting.
- c. Jason noted that he would like to see some 3-way (or N-way, with N greater than 2) included in this list.
- d. Frank noted that he wanted to mine some past ICDs as the basis of including interface types and data in these patterns.

12. Interface Work by Others:

- a. Marc Sarrell, NASA, was one of the key authors of the NASA/OMG papers that we had discussed last meeting. Marc looked up Bill Schindel at the ASME Model VVUQ

Conference both attended this week in Las Vegas. Bill discussed the Interface Patterns Project with Marc, and he is happy to help if needed.

- b. Frank will send Bill a copy of a model V&V document that may be related to the model VV&Q conference.
13. Priority project tasks within charter, as previously discussed:
 - a. Focus initially on examples Power, Mechanical (space reservation & mounting), & Data, illustrating how diverse interfaces can be well represented within a single specialize-able pattern framework, organized in its specialization by some taxonomy
 - b. Strong interest in generating model query view equivalent of an Interface Control Document (ICD) that is as complete as legacy ICDs, but more effective/efficient/uniform.
 - c. Discussed identifying some deliverables to have available by the time of IW2018, of a nature likely to draw others into this project and its application.
14. Plans for next activities:
 - a. Next meeting: Friday, May 19, 9:00 AM EST
 - b. Actions before next meeting: See Action Items

Action Items:

15. Generate list of highest priority interface types for initial patterns, exchange by email before next meeting (all team members).
16. Project tasks, potential assignments, schedule—either in Project Charter or equivalent (Bill)
17. Send out repeating meeting calendar invitation (Bill)
18. Post minutes and materials to project web site (Bill)

Reference Materials: (These may be down loaded from the following project web site)

19. Project web site: http://www.omgwiki.org/MBSE/doku.php?id=mbse:patterns:interface_patterns_team
Pages easily added under that location
Files can be inserted there when appropriate (see same)
20. Agenda, 05.05.2017 meeting of Interface Patterns Team
21. Interface Patterns Project Charter, INCOSE Patterns Working Group, V1.3.2
22. Abbreviated S*Metamodel Glossary, V4.3.1, ICTT System Sciences, 2017.
23. Extract from S*Metamodel: Interface Related Elements, 03.07.2017, V1.2.6, ICTT
24. Torok, J., Sherey, J., “Interface model content emails Feb-Mar 2017”.
25. Shames, Sarrel, Friedenthal, “A Representative Application of a Layered Interface Modeling Pattern”, Proc. of INCOSE International Symposium 2016, Edinburgh, UK; paper + slides:
http://www.omgsysml.org/A_modeling_pattern_for_layered_system_interfaces-INCOSE%20IS15_paper-sarrel-shames.pdf
http://www.omgwiki.org/OMGSysML/lib/exe/fetch.php?media=sysml-roadmap:a_representative_application_of_a_layered_interface_modeling_2016-07-11.pptx
26. Shames, Sarrel, Friedenthal, “Modeling Systems of Systems Interfaces with SysML”, AIAA 2016 Conference, SpaceOps Conference, (AIAA 2016-2500) <http://dx.doi.org/10.2514/6.2016-2500>
27. OMG SysML 2.0 Interface Concepts Team web site:
http://www.omgwiki.org/OMGSysML/doku.php?id=sysml-roadmap:interface_concepts_modeling_core_team
28. JPL MBEE Project wiki: <https://github.com/Open-MBEE/ProjectWiki>
29. Schindel, Extract from Requirements for Models Project, INCOSE Patterns Working Group and ASME, INCOSE MB Transformation, and VV50 Model Life Cycle Working Group, INCOSE Patterns, 2017.