

William Schindel

From: Jason Sherey <sherey@icct.com>
Sent: Monday, March 13, 2017 11:14 AM
To: 'Torok, Jonathan CIV NSWC Crane, GXTP'; 'William Schindel'; Frank.Salvatore@engilitycorp.com
Cc: 'Stephen Lewis'
Subject: RE: [Non-DoD Source] Interface Patterns Project--logistics, materials, etc.

Jon,

I am pleased to see this group get started on some in-depth discussion and modeling!

I can see why you chose an Association Block to model a Functional Interaction. A Functional Interaction is defined as a relationship between two or more systems, which would seem to map directly to a UML/SysML association, which is typed by an Association Block. I also agree that an Association Block, as all Blocks do, have structures useful to modeling the details of a Functional Interaction, including IBD diagrams, item flows, etc. I have run into a critical issue in pursuing this mapping strategy: SysML Associations are limited to two roles. Systematica allows more than 2 roles to interact within a Functional Interaction. Many of the Functional Interactions we model have more than 2 roles, especially in the manufacturing patterns where the equipment is combining two or more parts into a larger part. While it is true that a Functional Interaction can be decomposed into smaller 2-rolled Functional Interactions, we haven't typically done that because the larger one is usually the better descriptor of the physics that is occurring.

I have recently discovered that an Activity is also a type of Block. This means it also has the IBD diagrams, item flows, and other structures you mentioned. By being an Activity Block, it would also bring along optional Activity diagrams, which tie in to the EFFBD diagrams some SE's use. The actions that occur within the Activity seem to map into Systematica's Requirement "Transfer Function" concept of being a relationship between a systems inputs, outputs, and attributes. An Activity Block can aggregate more than 2 roles and its Activity Diagram can have more than 2 swim lanes that represent allocations of the action requirements to those roles. In addition, having a Functional Interaction mapped to an Activity Block enables SysML states to have do\Operations that call the behavior described by a Functional Interaction Activity. This State-Functional Interaction relationship has been a difficult one to map into SysML. In fact, I have also used the idea of a Functional Interaction owning a Functional Interaction Activity so that states can call the Functional Interaction. With Activities actually being a type of Block, we would no longer have to model two SysML elements to cover the concept of a Functional Interaction.

While what I just described has many parts that "feel right," the whole thing still seems too complex. I think the ability to easily and quickly model the Interaction-Role-Requirement relationship quickly without extra classes and other model elements is at least just as important as being able to explicitly model the input-output transformations and the relationships to the role's attributes. The example Interface models we make together should include a Domain Diagram and Functional Interactions that help tease out and explain the differences between a Systematica Functional Interaction that often has more than 2 roles and a simpler, more common definition of interaction that describes the interfacing between two systems. This is a very common point of confusion that is hard to avoid until there is a problem.

I look forward to a robust discussion and exciting progress in both the modeling strategies and explanations of those strategies!

Jason Sherey

-----Original Message-----

From: Torok, Jonathan CIV NSWC Crane, GXTP [mailto:jonathan.torok@navy.mil]

Sent: Thursday, March 09, 2017 9:14 AM

To: William Schindel; Frank.Salvatore@engilitycorp.com; sherey@icct.com

Cc: 'Stephen Lewis'

Subject: RE: [Non-DoD Source] Interface Patterns Project--logistics, materials, etc.

Bill,

1. Thanks for the charter. I like that it stays abstract. My only comment is that it should be NSWC Crane as opposed to Crane NSWC.

2. I received a username and password, and it works. Do you want "draft" material posted here? If not, can we use something as simple as dropbox or google drive?

3/4/5. I think these should be topics of discussion for a first meeting.

I've been digesting your material, but I have some questions that will probably need an example shown. The largest difference to my work was use of an association block. I used a "functional interaction" stereotype on an association block to group all elements of an interface/interaction.

Function(s), IBD structure, SOAs, Item flows etc. can all be properties of that association block. Attached is an example from our modeling. It doesn't matter if Functional Interaction is the stereotype of the association block, but that was the type of assumption I made from the S*pattern documentation. However, I would assume that "functional interaction" is the entirety of the interaction, where as a name such as "interaction function" would be the actual activity of an interaction.

6. I'm flexible on scheduling time, but would prefer avoiding the weekend.

This might be obvious, but the OOSEM WG meets on Saturday mornings.

Additionally, I am currently plagued by not having any screen sharing service work on my work computer, even the government issued one.... Since we are both in Indiana, I am open to traveling to your location so that it will be easier to collaborate with you and Frank.

Jon Torok

-----Original Message-----

From: William Schindel [mailto:schindel@icct.com]

Sent: Tuesday, March 07, 2017 1:02 PM

To: Torok, Jonathan CIV NSWC Crane, GXTP; Frank.Salvatore@engilitycorp.com; sherey@icct.com

Cc: 'Stephen Lewis'

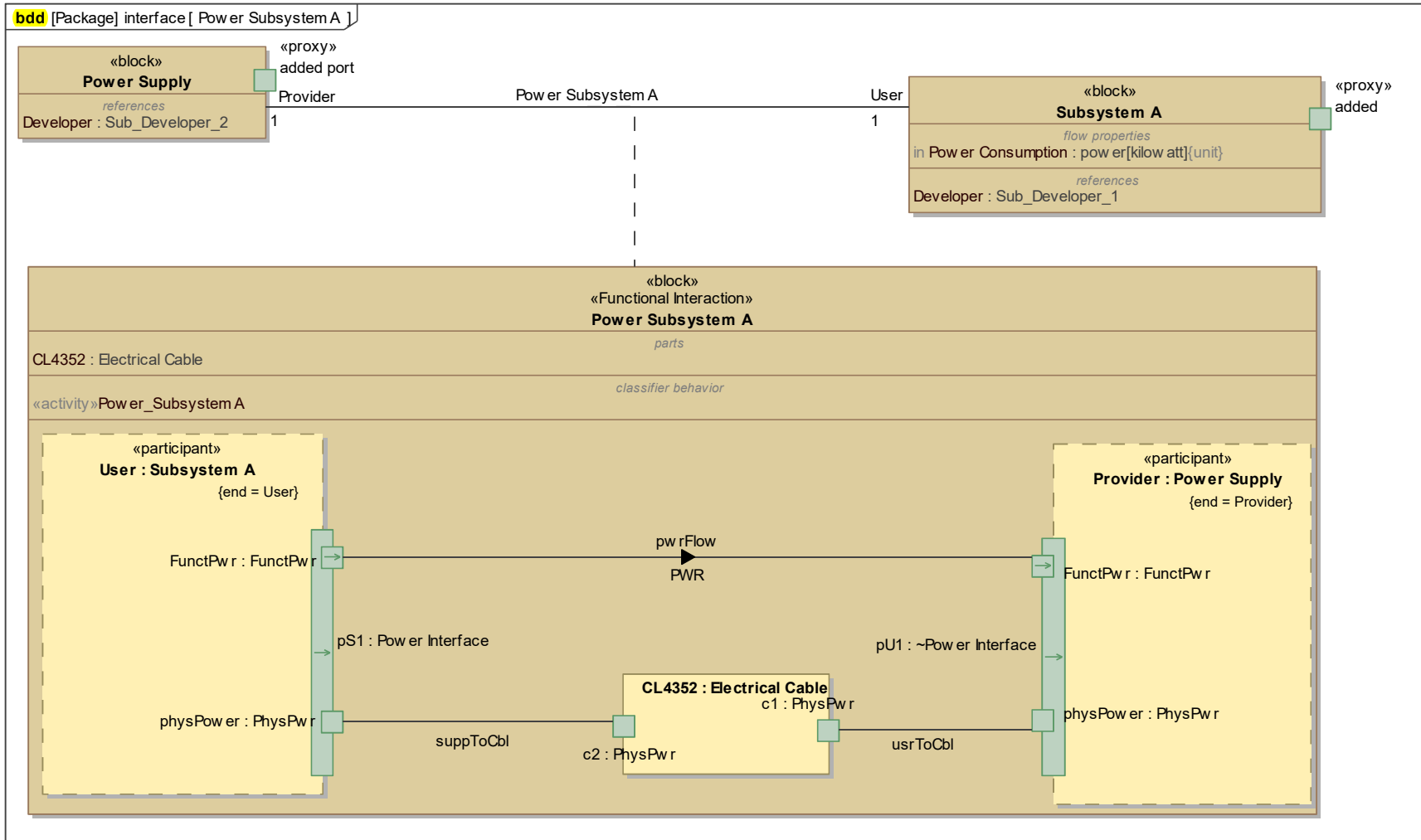
Subject: [Non-DoD Source] Interface Patterns Project--logistics, materials, etc.

Jonathan, Frank, and Jason,

Finally getting the Interface Patterns Project support in place:

1. Project Charter: Attached is a draft project charter, for your review, improvement, and feedback. Please advise on what needs to change, ideas to fill in aspects, questions and concerns, etc.

2. Project Web Site: A web site for posting work products, references,



William Schindel

From: Torok, Jonathan CIV NSW Crane, GXTP <jonathan.torok@navy.mil>
Sent: Friday, March 03, 2017 1:56 PM
To: Frank.Salvatore@engilitycorp.com; William Schindel
Cc: sherey@icct.com; Chapman, Jacob A CIV GXTM
Subject: RE: [Non-DoD Source] RE: Interface Patterns

Frank,

We went down this path some here at Crane doing some experiments. I was hoping the share space Bill mentioned would be up to start sharing.

If we are going down the SysML path, I think the first item to do is lock down the S*pattern stereotypes, and what elements of an interface that they will apply to. Bill had also mentioned being able to share a S*pattern profile. I wanted to make sure I wasn't stepping on it before showing how we wanted to apply the S*Pattern definitions to types of SysML elements.

You started to classify below. So far, we have attempts to do a power interface and a bolt pattern for this, but neither are a in great shape to share. I can try to get out some PDF captures early next week. We chose these first because they are fairly simple. We have been attempting to use a pattern to create a repeatable generic table in MagicDraw. The pattern helps, but we are having difficulties with the metachain tools in MagicDraw. I'm trying to get help from No Magic, but some of our issues may require development from No Magic. There is probably some benefit to different groups trying to model a single type of interface.

We have also thought about how to abstract an interface to be able to build it as more details become available. This leaves some interfaces only being functional for a time. To accomplish this type of activity, we are looking at what JPL has shown. I also found some work that OMG is doing that is an extension of JPLs interface patterns: http://www.omgwiki.org/OMGSysML/doku.php?id=sysml-roadmap:interface_concepts_modeling_core_team. These patterns tie into expanding on how we looked at modeling the "system of access" from S*patterns. We might want to collaborate with their efforts, or in the least let them know of our efforts.

Lastly, do we want to agree on a version of MagicDraw? 18.5 is out. A preview version of 19 is out, but we cannot install it here yet.

Jon Torok

-----Original Message-----

From: Frank.Salvatore@engilitycorp.com [mailto:Frank.Salvatore@engilitycorp.com]
Sent: Thursday, March 02, 2017 2:35 PM
To: William Schindel; Torok, Jonathan CIV NSW Crane, GXTP
Cc: sherey@icct.com
Subject: RE: [Non-DoD Source] RE: Interface Patterns

How about we classify the Interface types and determine if they warrant different S Patterns. Then once we do that we can work on one and see if it works. Then we will have a pattern for making Interface patterns J.

I would classify Interfaces in the following way:

- . Physical (e.g. coupling)
 - * Mechanical Interfaces
 - * Fluid sealing interfaces
- . Logical (e.g. data)
- . Electrical (e.g. power)
- . Energy
- . Computer User Interfaces
 - * command line.
 - * graphical user interface (GUI)
 - * menu driven.
 - * form based.
 - * natural language.

From: William Schindel [mailto:schindel@icct.com]
Sent: Thursday, February 09, 2017 5:02 PM
To: 'Torok, Jonathan CIV NSW Crane, GXTP'; Salvatore, Frank @ EngilityCorp
Cc: sherey@icct.com
Subject: RE: [Non-DoD Source] RE: Interface Patterns

EXTERNAL EMAIL -- This message originates from outside of Engility.

Hi Jon,

Glad to see your thoughts below. There is indeed an S*Metamodel profile for SysML, in fact there are specializations of it, as very similar profiles for Magic Draw, Enterprise Architect, and Rhapsody. We are working on ways to make these available, and I would like to do so for this WG project. More on that shortly. Are you interested in using Magic Draw /Cameo System Modeler?

You are correct to refer to the portion of the S*Metamodel which is particularly about interfaces, related to the attachments sent to you Feb 2 below. One could consider that subset of the S*Metamodel to be the most abstract S*Pattern for Interfaces, since it describes all interfaces, prior to specialization and configuration to their specific cases. There would then be a bunch of specializations of that, into S*Patterns of interfaces of different sorts, still relatively general, but more specific than the most abstract one (the S*Metamodel level) and conforming to it. I would think the working group project might be viewed as

William Schindel

From: Torok, Jonathan CIV NSWCR Crane, GXTP <jonathan.torok@navy.mil>
Sent: Friday, February 10, 2017 8:51 AM
To: William Schindel; Frank.Salvatore@engilitycorp.com
Cc: sherey@icctt.com
Subject: RE: [Non-DoD Source] RE: Interface Patterns

Just a short reply (before the thought leaves my brain),

ICDs can refer to a single system's interface or both sides of an interface. For work here at Crane, we more typically have ICDs of the latter. Looking back over the PBSE WG material, I wanted to point out slide 28 of: http://www.omgwiki.org/MBSE/lib/exe/fetch.php?media=mbse:patterns:pbse_wg_meeting_slides_jan_31_2016_v_1.2.1.pdf. I don't believe that view was in the previous material Bill sent. The ICD of the Functional Interaction with interface 1/2 and the system of access is of great interest to my work.

MagicDraw/Cameo is what I have access to use.

Jon

-----Original Message-----

From: William Schindel [mailto:schindel@icctt.com]
Sent: Thursday, February 09, 2017 5:02 PM
To: Torok, Jonathan CIV NSWCR Crane, GXTP; Frank.Salvatore@engilitycorp.com
Cc: sherey@icctt.com
Subject: RE: [Non-DoD Source] RE: Interface Patterns

Hi Jon,

Glad to see your thoughts below. There is indeed an S*Metamodel profile for SysML, in fact there are specializations of it, as very similar profiles for Magic Draw, Enterprise Architect, and Rhapsody. We are working on ways to make these available, and I would like to do so for this WG project. More on that shortly. Are you interested in using Magic Draw /Cameo System Modeler?

You are correct to refer to the portion of the S*Metamodel which is particularly about interfaces, related to the attachments sent to you Feb 2 below. One could consider that subset of the S*Metamodel to be the most abstract S*Pattern for Interfaces, since it describes all interfaces, prior to specialization and configuration to their specific cases. There would then be a bunch of specializations of that, into S*Patterns of interfaces of different sorts, still relatively general, but more specific than the most abstract one (the S*Metamodel level) and conforming to it. I would think the working group project might be viewed as especially focused on those more specific types, but I also agree that the subset of the S*Pattern which is about Interface ideas should also be a key part of the project, with which we'd like to "prime" the project, representing what we have found to be the most general interface pattern.

In listing what we'd like to do with such patterns (as you have below), I would add that folks typically want to generate Interface Control Documents (ICDs) or the like. Existing legacy ICDs thus become a good real world test of the pattern.

The above also brings up a classification question, which the attached drawing illustrates. I pulled it from a generalized manufacturing system pattern we use, which includes a family of interfaces that occur in the manufacturing world, including mechanical, electronic, and otherwise. (The "System of Access" is just part of the Interface metamodel--don't be distracted by it for now.) The question I am referring to is: how would we like class to classify / organize families of interface patterns, under the most abstract one. The attached taxonomy is only one of many ways. It reflects a bias toward the type of physical system of access involved, but might not be how you see these families.

Something to consider while I am fumbling with the administrative aspects to get started

Bill

-----Original Message-----

From: Torok, Jonathan CIV NSWC Crane, GXTP [mailto:jonathan.torok@navy.mil]

Sent: Monday, February 06, 2017 9:22 AM

To: William Schindel; Frank.Salvatore@engilitycorp.com

Subject: RE: [Non-DoD Source] RE: Interface Patterns

Thanks for the additional info Bill.

a) From your (2.), my thoughts are to get to implementation. I would like to implement interfaces using S*Metamodel in SysML (SysML being the current favorite flavor of MBSE). From there, creating actual templates and examples of specific interfaces should provide response back to S*Metamodel as validation or improvements. In the SysML "experiments" I've been doing, the patterns have been similar to the S*metamodel, but I have not implemented any intelligent stereotype/metamodel rigor yet. This experimentation is done at the low level with particular types of interfaces, but at this point, I want to go up to a higher level abstract pattern for interfaces to ease making the metadata parsable.

b) Is there an S*Metamodel SysML profile? If not, should we create one to use, or just focus on the interface portions of the S*metamodel at this point?

Regards,

Jon Torok

-----Original Message-----

From: William Schindel [mailto:schindel@ictt.com <mailto:schindel@ictt.com>]

Sent: Thursday, February 02, 2017 3:59 PM

To: Frank.Salvatore@engilitycorp.com <mailto:Frank.Salvatore@engilitycorp.com> ; Torok, Jonathan CIV NSWCR Crane, GXTP

Subject: [Non-DoD Source] RE: Interface Patterns

Hello Jonathan and Frank,

Sounds like a good potential collaboration is brewing! J I knew Frank has had this interest a long time, and I was happy to learn that my fellow Hoosier Jonathan also has it.

Let me offer some support, and also one constraint important to us, both as related to the INCOSE/OMG MBSE Patterns Working Group:

1. As quickly as I can clean off my "desk" from IW, I will offer you some support in the way of a more detailed S*Pattern example representation of an interface, and I will be happy to answer questions, review drafts, provide feedback, or help you otherwise, as long as it looks to you like help and not interference. Meanwhile, attached are some preparatory items to help with a few interface-related S* subjects.

2. A "price" of doing this under the umbrella of the INCOSE Patterns WG is to learn and use the corresponding part of the S*Metamodel, which is simple enough and I hope you will agree adds value, not obstacles. Pending selection of a detailed example to send you, I have attached an highlighted a few of the interface-related S*Metamodel definitions. Conformance to