**GRA-UML**

**Design Document**

**UML & BPMN**

3/31/2014

# Overview

This document is intended to help facilitate a discussion on the role BPMN plays in GRA-UML. The GRA community has an established track record with BPMN yet the profile is based on UML. There are “shares of grey” as to how much it is UML Vs. BPMN and tradeoffs either way. The RFP requires UML,. SoaML and BPMN – it is not clear what it means to have all of these together.

# RFP Requirements

The RFP requires that the resulting profile be based on elements from UML, SoaML & the UML profile for BPMN as highlighted in yellow below. It also requires that it output GRA specifications as outlined it blue.

**6.5 Mandatory Requirements**

6.5.1 Submissions shall specify a GRA-UML Logical Profile. The GRA-UML Logical Profile shall be a set of UML stereotypes and properties which support the modeling of SSPs in UML in a technology-independent way. This profile shall support modeling of any content and structure allowed by GRA SSPs, while constraining the modeling of any content and structure disallowed by GRA SSPs, as specified in 6.5.4. The use of the GRA-UML Logical Profile shall result in UML models that are free from dependency on any physical representation (such as XML Schema). In MDA terms, the GRA Logical Profile is a specification of the platform independent model (PIM).

6.5.3 Submissions shall specify a transformation from UML models using the GRA-UML profiles specified in 6.5.1 and 6.5.2 to the set of artifacts required in a conformant SSP, as defined by 6.5.4. Submissions shall utilize the GRA-UML Profiles to model at least one existing GRA SSP and demonstrate the resulting transformation to an SSP. The SSP produced must be GRA conformant, as defined in section 6.5.4, and the XML Schema set contained within the SSP must validate the same set of exchange documents as the existing SSP IEPD. It is not required that the generated SSP be structurally identical**1** to the existing SSP.

6.5.4 The SSPs generated based on models conforming to the GRA-UML profile shall conform to normative GRA specifications referenced in section 6.4.1. These specifications are:

 GRA Service Specification Package, v1.0.0 (http://it.ojp.gov/docdownloader.aspx?ddid=1217)

 GRA Service Specification Guideline v1.0.0 (<http://it.ojp.gov/docdownloader.aspx?ddid=1215>)

OMG RFP - UML Profile for GRA (GRA-UML) – 23-September-2013 27

 GRA Web-Services Service Interaction Profile v1.3 (http://it.ojp.gov/docdownloader.aspx?ddid=1173)

 GRA ebXML Messaging Service Interaction Profile v1.1 (http://it.ojp.gov/docdownloader.aspx?ddid=1168)

6.5.5 GRA Reliable Secure Web Services, Service Interaction Profile v1.2 (<http://it.ojp.gov/docdownloader.aspx?ddid=1134>). Specifications shall reuse constructs and/or representations from SoaML to express the SSP logical model and will extend SoaML to meet specific requirements of the GRA where required.

6.5.6 Specifications shall reuse elements and/or representations from UML Profile for BPMN 2 Processes to express the SSP logical model and more specifically the SSP interaction model.

## GRA Requirements

The GRA “Service Specification Package Artifacts” specifies the inclusion of a process model “for the web service” as follows:

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Business Process Model | This is the actual document which describes the business process model for the Web service. In many cases, this can be used to import/export the process model for the service. | BPMN, BPEL, JIEM, UML | R |  |

Note that there are multiple options for the process model format. However, BPMN has emerged as the preferred diagram format.

The EbXML SIP also specifies use of “OASIS ebXML Business Process Specification Schema v2.0.4 [ebBP]” which has capabilities similar to SoaML and parts of BPMN.

# Relationship between UML, BPMN and the UML profile for BPMN

UML and BPMN-2 are both OMG standards and both are based on the OMG modeling infrastructure, MOF. However they are independent metamodels and not directly related. UML contains “activity diagrams” and “Collaboration diagrams” that are similar in capability to parts of BPMN. A process in a UML activity diagram will look almost identical to a BPMN process to the average stakeholder. While similar in capabilities the “metamodels” of BPMN and UML are not related (this is a known and persistent problem within OMG). The” UML profile for BPMN” maps these meta models, however it is not a 100% mapping in either direction.

SoaML is a “thin profile” based on UML that adds serve specific stereotypes. SoaML has stereotypes that focus at the business layer (collaborations and participants) and also on the service implementation layer (interfaces and components). SoaML uses regular UML features for service choreography – either activity diagrams or (at a lower level) sequence diagrams.

BPMN’s strong point and origin is the internal process of an organization. As of BPMN-2 more features were added for service modeling – collaborations, conversations and Choreography tasks. These service modeling features were influenced by SoaML but BPMN did not include all of the service modeling features. BPMN-2 is capable of but probably not the best choice for detailed service interfaces and interactions.

UML has class diagrams that provide for interfaces, classes and ports. It also has interaction diagrams that are most used for choreography and activity diagrams for business processes. As such UML, particular when coupled with SoaML, has most of the features of BPMN that would be relevant to GRA. UML is particularly good at interfaces which are the essential elements of WSDL. Due to this overlap it must be decided what will be natively in UML and/or BPMN.

What is not covered well by any of the above is service policies, so we will leave this out of the discussion for now.

# Kinds of process

There are different layers and abstractions of processes as follows:

## The traditional internal business processes

When most people think of a business process they think of the internal activities of an organization. Such a business process defines what an organization does which then leads to how it provides products and services as well as to what resources it consumes. AN internal process is less focused on external interactions.

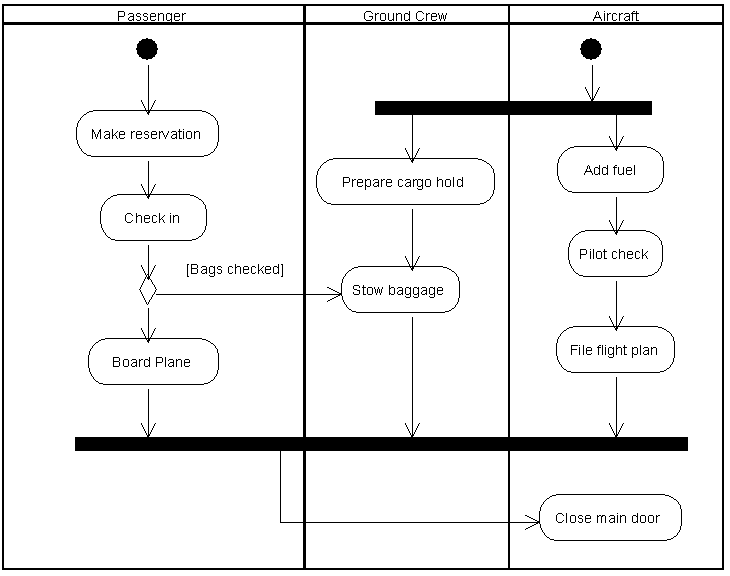
A traditional business process can be specified at a high-level, primarily for stakeholders to get a “rough idea” what the business does and how it may be improved. A business process can also be specified at a detailed level such that automation can help realize and execute the process (e.g. a BPM engine).

From an external process perspective in BPMN, messages arrive from external participants and are sent to them – the “conversation” they have with external parties are a derivative of their internal process – which is not particularly appropriate for an external services agreement, thus some of the BOMN-2 additions.

Internal processes are the central focus of BPMN process and UML activity diagrams.

### Random BPMN Example

Random UML Example



## Collaborative business processes

A collaborative business process defines how a community of independent entities work together for some shared goal – e.g. for commerce or stopping crime. Individual organizations play a role in such a collaborative process and interact via services. The internal processes and activities of the involved organizations are usually hidden from each other with the philosophy that “we don’t care how you do it, just so that you fulfill your responsibilities to the community”. A strongly “encapsulated” collaborative business process completely hides internal processes and activities. Less encapsulated “open” collaborative processes expose some required process steps of the participants.

Since the strongly encapsulated collaborative processes expose no internal activities the only thing any entity knows about the overall process is what they see via services. There is no overall “controller” or “execution environment”, only the controllers and execution environments of the participants. If decisions are made based on activities within an entity that impact another entity (without a service notification) there has to be some overall process controller – which is generally thought of as not best practice.

Like an internal process, a collaborative business process can be specified at a high-level, primarily for stakeholders to get a “rough idea” what members of the community do and the relevant business services. A business process can also be specified at a detailed level such that automation can help realize and execute the process (e.g. a BPEL engine).

Both BPMN and UML collaborations can be used for encapsulated collaborative processes. Both BPMN and UML can be a bit clumsy when used for non-encapsulated collaborative process (when the internal activities are shown), but BPMN may be a bit better as it is more focused on internal processes. UML is better at the services.

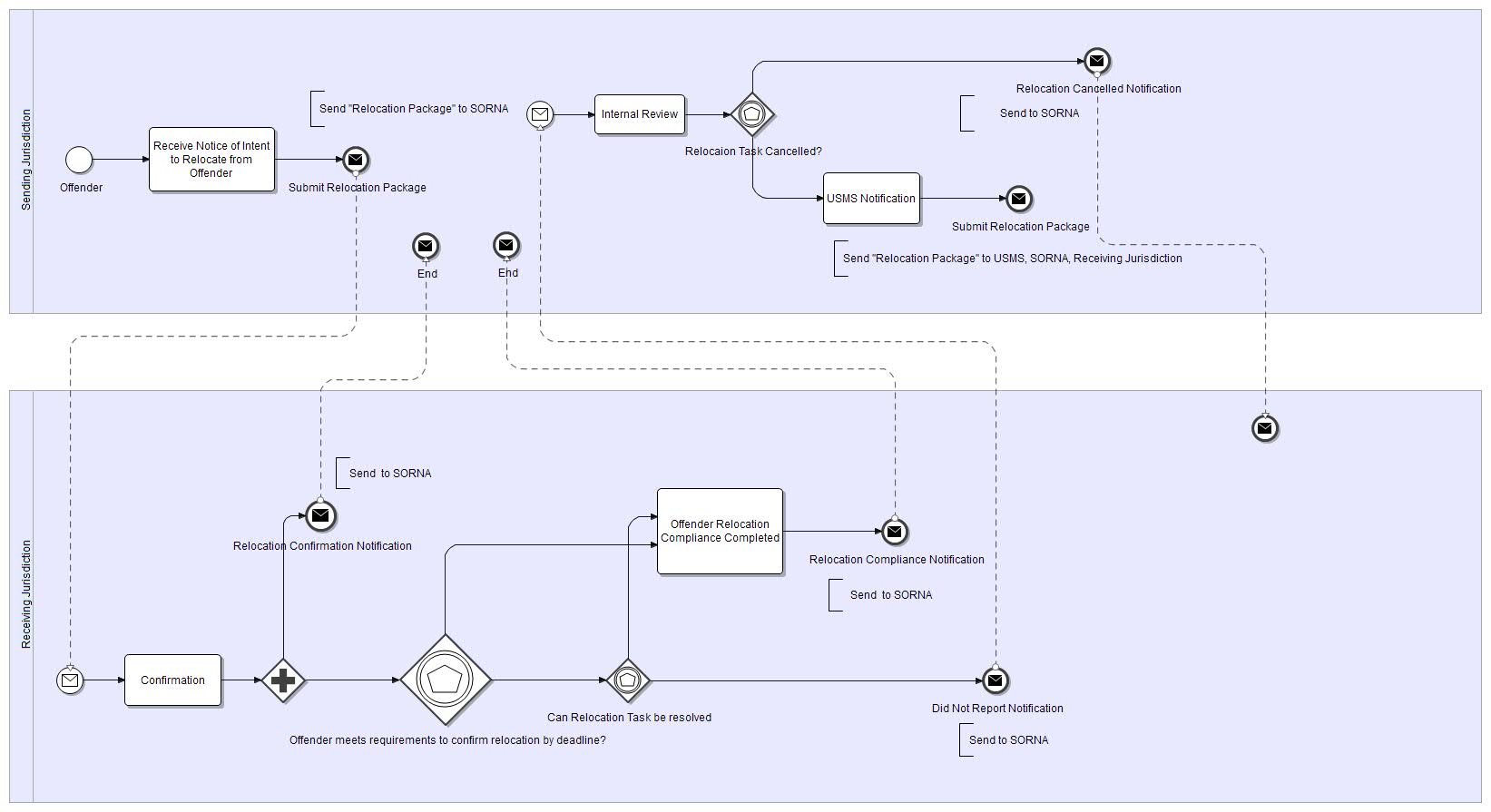
### Example of a fully encapsulated collaborative process (BPMN)

Note that in the diagram below we only see the participants and services (conversations in BPMN) that connect them.

### Example of a fully encapsulated collaborative process (UML/SoaML)

### Example of an open (not encapsulated) process (BPMN)

Note in the following diagram that activities internal top the participants is shown. The choreography of the conversation is implied based on the internal processes. The UML variant of an activity diagram would look very similar.



## Service interaction processes (Choreographies)

A “service” is two or more entities interacting (perhaps within a collaborative business process) to achieve some purpose. A service interaction process is much like a collaborative process but usually involving only a service “provider” and “consumer”. A service typically enables some kind of “business transaction” – an exchange of products, services, information, assets or obligations between the parties. A service can be broken down into specific “flows” of these products and services – in technologies each interaction is described by a technology service interaction (data going over the wire).

The “process” of a service interaction is the flows between the provider and consumer (as well as any auxiliary parties) – these flows are typically choreographed and may come with other requirements and obligations at both the business and technical level. The service interaction is also where the connection to specific data exchange elements is made – this data connection is very important but a weak point in BPMN.

The typical “web service” is essentially a service interaction. The context for a set of related service interactions is a collaborative business process.

Like an internal process, a service interaction process can be specified at a high-level, primarily for stakeholders to get a “rough idea” of the service exchange. A service interaction process can also be specified at a detailed level such that automation can help realize and execute the process (e.g. a SOA engine or ESB).

UML is very effective for specifying service interactions – interaction diagrams are usually favored but activity diagrams can also be used. BPMN has the “Choreography Task”, but there are issues with the connection of Choreography Tasks with data, conversations and collaborations.

### Example of a service interaction in BPMN

### Example of a service interaction in UML (Interaction and activity diagram style)

## Service Interfaces

While not technically a “process” service interfaces should also be mentioned as these are the primary element exposed in technologies as specified in a SIP. Interfaces define what can go into or out of a particular entity or technology component. SOA interfaces are typically bound to a “port” (a point of communication) on some participant or component.

UML is very good at specifying interfaces that are “one way” (information goes to an entity or uses request/reply). SoaML adds 2 way (or multi way) conversations, a concept not native to UML. Ports and classes and components are native to UML.

BPMN has no native notion of an interface; the interfaces are inferred from processes.

### Example of participants with service interfaces and ports (SoaML)



# Tooling considerations

While both BPMN-2 and UML are defined in the OMG modeling infrastructure, they are generally implemented with separate tools that do not work together. The UML profile for BPMN was intended to provide a mapping and a way for UML tools to implement parts of (but not all of) BPMN. The UML profile was driven by the UPDM (DoDAF) effort as UPDM is based on UML/SysML and they wanted to use BPMN.

It is possible to use both UML and BPMN (Native) tools and map information between them but such mappings are often a burden on both the process and pocketbook. It is easiest to stay in one tool.

One of the goals of the community is to have an open source implementation of GRA-UML. At this time we do not know of an open source implementation of the UML profile for BPMN. This UML profile is a “thick” profile with new diagrams, it is non-trivial to implement.

In addition, while the UML profile for BPMN implements much of BPMN, the parts missing have to do with services. The exact match with respect to GRA is TBD. Also, intermixing UML and BPMN elements is experimental.

EbXML also has its own concept of collaborative and choreographed processes, and its own set of tools.

In summary, it would be easiest to stay within one tool and use native capabilities as much as possible.

# The Choices

For the above reasons we need to be careful about how we use and intermix UML+SoaML & BPMN. In that their capabilities are probably a 90% overlap for services modeling either a “mostly UML” or “Mostly BPMN” direction could be made to work.

If “Mostly BPMN” the question should be asked – should we instead do a Profile of BPMN and only use UML for the data? In this scenario we would have GRA-BPMN and probably re-issue the RFP.

If “Mostly UML” the question is, should we really use BPMN at all? Or, should we just generate some BPMN “hooks” to help with high-level stakeholder diagrams? Or, BPMN could be looked at as requirements and not impact the service implementations.

Since the UML profile of BPMN is specified in terms of UML and a mapping to BPMN, we could specify in terms of UML and leave it to tools to decide how to marry the two.

If split, we should then look at each kind of process and see how it is represented in each and what the relationship is:

* Internal processes (out of scope?\*)
* Collaborative processes (encapsulated and open)
* Service interaction choreographies
* Service interfaces

If split we have to be very careful about the practicality of using the different tools and or perspectives in tools.

\*We would question the applicability of internal processes for GRA-UML at all, while it is highly desirable for implementers of a service to connect internal processes with services – the service philosophy says that the service specification should NOT specify ANY part of the internal service implementation or process. However, internal processes are frequently used to help define the requirements for collaborative processes.

Another choice would be to allow use of either UML or the UML profile for BPMN, this would require 2 ways to do the same thing in the same specification. While this has been done, it is not ideal.

Our choice would be to focus on UML and a thin profile on top of it. We would then view BPMN as “auxiliary” and used for high-level process diagrams that would not directly impact the service specification. Your mileage may differ.