NATO Communications and Information Agency
Agence OTAN d'information et de communication

AGENCY INSTRUCTION

INSTR TECH 06.02.08

SERVICE INTERFACE PROFILE FOR PUBLISH-SUBSCRIBE SERVICES

Effective date:

Revision No: Original

Issued by: Chief, Core Enterprise Services _____

Approved by: Director Service Strategy _____

Table of Amendments

Amendment No	Date issued	Remarks

Author Details

Organization	Name	Contact Email/Phone
NCI Agency	V. de Sortis	vincenzo.desortis@ncia.nato.int
NCI Agency	M. Lehmann	marek.lehmann@ncia.nato.int
NCI Agency	R. Fiske	rui.fiske@ncia.nato.int
NCI Agency	L. Schenkels	leon.schenkels@ncia.nato.int
NCI Agency	G. Gujral	davinder.gujral@ncia.nato.int

Table of Contents

AGENCY INSTRUCTION 06.02.08

SERVICE INTERFACE PROFILE FOR PUBLISH-SUBSCRIBE SERVICES

0   PRELIMINARY INFORMATION

0.1   References

A. NCIA/GM/2012/235; Directive 1 Revision 1; dated 3 May 2013
B. NCIARECCEN-4-22852 DIRECTIVE 01.01, Agency Policy on Management and Control of Directives, Notices, Processes, Procedures and Instructions, dated 20 May 2014
C. NCIARECCEN-4-23297, Directive 06.00.01, Management and Control of Directives, Processes, Procedures and Instructions on Service Management, dated 03 June 2014

0.2   Purpose

This Technical Instruction (TI) provides detailed information, guidance, instructions, standards and criteria to be used when planning, programming, and designing Agency products and services. In this specific case the TI defines a Service Interface Profile (SIP) for one of NATO's Core Enterprise Services.

TIs are living documents and will be periodically reviewed, updated, and made available to Agency staff as part of the Service Strategy responsibility as Design Authority. Technical content of these instructions is the shared responsibility of SStrat/Service Engineering and Architecture Branch and the Service Line of the discipline involved.

TIs are primarily disseminated electronically[1], and will be announced through Agency Routine Orders. Hard copies or local electronic copies should be checked against the current electronic version prior to use to assure that the latest instructions are used.

0.3   Applicability

This TI applies to all elements of the Agency, in particular to all NCI Agency staff involved in development of IT services or software products. It is the responsibility of all NCI Agency Programme, Service, Product and Project Managers to ensure the implementation of this technical instruction and to incorporate its content into relevant contractual documentation for external suppliers.

1   SIP INTRODUCTION

Publish/Subscribe (or pub/sub) is a messaging pattern where consumers may be dynamically subscribed to receive *notifications* from producers. A *subscription* indicates the consumer's interest in one or more message types. Consumers only receive messages that are of interest. The consumers may receive notification messages from producers directly or indirectly via *notification brokers*.

The publish/subscribe-based services are an important part of the NATO Network Enabled Capability (NNEC) Core Enterprise Services (CES) as defined in the CES Framework [NAC AC/322(SC/1)N(2009)0015 (INV), 2009]. The CES Implementation Guide [NC3A RD-2814, 2009] recommends [OASIS WS-BaseNotification, 2006] and [OASIS WS-BrokeredNotification, 2006] as the standards to implement a simple object access protocol (SOAP)-based publish/subscribe service in NATO.

---

[1] https://servicestrategy.nr.ncia/SitePages/Agency%20Directives%20(Technical).aspx

This document gives directives along with clarifications and amendments to the [OASIS WS-BaseNotification, 2006] and [OASIS WS-BrokeredNotification, 2006] specification on how to implement a *notification broker/subscription manager* to promote interoperability between the publish/subscribe engines and generic message subscribers. Some extensions to the protocol have been introduced in order to meet NATO requirements.

It is important to stress that the application of this Service Interface Profile (SIP) is a necessary but not sufficient condition to achieve service interoperability because it intentionally does not contain details of the domain-specific payload sent in notifications. To achieve a complete interoperability such details must be agreed between the *notification producer* and the *notification consumer* to guarantee the correct interpretation of the notification message.

1.1 Audience

The target audience for this specification is the broad community of NNEC stakeholders, who are delivering capability in an NNEC environment, or anticipate that their services may be used in this environment.

These may include (but are not limited to):

- Project Managers procuring Bi-SC (Bi-Strategic Commands) or NNEC related systems
- The architects and developers of service consumers and providers
- Coalition partners whose services may need to interact with NNEC services
- Systems integrators delivering systems into the NATO environment.

1.2 Notational Conventions

The following notational conventions apply to this document:

- The keywords "MUST", "MUST NOT", "REQUIRED", "SHALL", "SHALL NOT", "SHOULD", "SHOULD NOT", "RECOMMENDED", "MAY", and "OPTIONAL" in this document are to be interpreted as described in [IETF RFC 2119, 1997].
- Words in *italics* indicate terms referenced in Section 1.3.

1.3 Terminology

The following terminology is used in this SIP and in the related SIPs [NCIA AD 06.05.04.02.G], [NCIA AD 06.05.04.02.F].

Table 1 Terminology

Notification	A one-way message sent to indicate that an event has occurred.
	It is represented as an XML (extensible mark-up language) element with a Namespace qualified QName and a type defined using XML Schema.
	It is application-domain-specific.
Notification Producer	A web service capable of producing notifications for those *notification consumers* for which *subscriptions* have been registered, based on events that occur and on the parameters supplied with the requests from which the *subscriptions* were created.
	It may directly produce notifications itself, or it may be a *notification broker*,

	reproducing notifications that were produced by a separate *publisher* and/or *notification producer* entity.
Notification Consumer	An end-point, represented by a WS-Addressing end-point reference, designated to receive notifications produced by a *notification producer* as a result of a *subscription*. It may accept a generic notification message, or it may be able to process one or more domain-specific notification types.
Subscriber	Any entity that can create and manage *subscription*s interacting with *notification producer* and *subscription manager* services. Note that a *subscriber* may be a different entity from the *notification consumer* for which *notifications* are actually produced.
Subscription	Relationship between a *notification consumer* and a *notification producer*, including any filtering parameters such as *topic* and various other optional filter expressions, along with any relevant policies and context information.
Topic	Categorization that can be attached by a *notification producer* to a *notification*. A *notification* can be associated with one or more *topics*. *Topics* can be used by a *notification producer* to determine which subscribed *notification consumer*s should receive the *notification*.
Push-Style Notification	Occurs when the *notification producer* initiates a message exchange to send the *notification*.
Pull-Style Notification	Occurs when the *notification consumer* initiates a message exchange to receive the *notification*.
Passive Notification Consumer	A *notification consumer* that receives a *notification* following the "push-style" notification pattern.
Notification Broker	Web service that acts as an intermediary between *notification consumer*s and *publishers* in order to permit the *notification consumer* to subscribe to *notifications* produced by *publishers* that are not offering the *subscription* interface. In this role it acts as *notification producer* and *notification consumer* and can offer the interface for the *publisher registration*.
Subscription Manager	A *subscription manager* provides operations that allow a service requestor to query and manipulate *subscriptions* that it manages. A *subscription manager* is subordinate to the *notification producer/notification broker*, and may be implemented by the *notification producer/notification broker* service provider. However, it is permitted for it to be implemented by a separate service provider, should an implementer so desire.

Publisher	An entity capable to produce *notifications* sent to subscribed *notification consumers* or to a *notification broker* for further distribution to subscribed *notification consumers*. A *publisher* does not necessarily have to be a web service. A *publisher* may not be able to receive and process *subscriptions*. A *publisher* may be registered with a *notification broker* as capable of publishing notifications on given *topics*.
Pull Point	In the *pull-style notification* scenario it is an end-point usually created by a *subscriber* that accumulates notification messages and allows a requestor to retrieve accumulated notification messages.
Publisher Registration	Represents the relationship between a *publisher* and a *notification broker*, in particular, which *topic(s)* the *publisher* is able/permitted to publish.

1.4   Namespaces

The following namespaces are used in this SIP and in the related SIPs [NCIA AD 06.05.04.02.G], [NCIA AD 06.05.04.02.F]:

Table 2   Namespaces

Prefix	Namespace
wsa	http://www.w3.org/2005/08/addressing
wsnt	http://docs.oasis-open.org/wsn/b-2
wsn-br	http://docs.oasis-open.org/wsn/br-2
soap	http://schemas.xmlsoap.org/soap/envelope/ or http://www.w3.org/2003/05/soap-envelope
wsn-nato	http://nc3a.nato.int/ces/wsn/extension

1.5   Goals

This SIP is intended to give directives, along with clarifications and amendments on the use of required and recommended interfaces to be implemented by the *notification consumer* and *notification broker* together with *subscription producer* compliant with the [OASIS WS-BrokeredNotification, 2006] and [OASIS WS-BaseNotification, 2006] specifications.

1.6   Non-Goals

The following topics are outside the scope of this profile:

- Propose a new publish/subscribe specification.
- Define the circumstances under which a *notification* should be produced.
- Define the format and/or the semantics of the *notification.*
- Define the means by which *notification broker/notification producer* are discovered by *subscriber*s. It is beyond the scope of this SIP to define the mechanisms for run-time discovery of *notification brokers* and/or of *notification producers*.

1.7   Relationships to Other Profiles and Specifications

1.7.1   Normative References

WS-Notification is a family of related specifications that define a standard web-service approach to *notification* using a topic-based publish/subscribe pattern. It includes standard message exchanges to be implemented by service providers that wish to participate in:

- Notification-based messaging.
- Standard message exchanges for a *notification broker* service provider (allowing publication of messages from entities that are not themselves service providers).
- Operational requirements expected of service providers and requestors that participate in notifications.
- An XML model that describes topics.

This SIP is based on the following WS-Notification normative specifications, which MUST be followed in all the areas not covered in this document and in [NCIA AD 06.05.04.02.G], [NCIA AD 06.05.04.02.F].

1.7.1.1   WS-BaseNotification 1.3

http://docs.oasis-open.org/wsn/wsn-ws_base_notification-1.3-spec-os.pdf

1.7.1.2   WS-BrokeredNotification 1.3

http://docs.oasis-open.org/wsn/wsn-ws_brokered_notification-1.3-spec-os.pdf

1.7.1.3   WS-Topics 1.3

http://docs.oasis-open.org/wsn/wsn-ws_topics-1.3-spec-os.pdf

2   PUBLISH/SUBSCRIBE FOR WEB SERVICES

Publish/Subscribe is a message-exchange pattern which minimizes the amount of unnecessary polling, where a data consumer periodically queries the producer for new events. Instead the consumer can be subscribed to receive notifications in response to new events.

The CES Implementation Guide [NC3A RD-2814, 2009] proposes WS-Notification 1.3 as the standard to implement a SOAP-based publish/subscribe service in NATO. WS-Notification is a family of related specifications: [OASIS WS-BaseNotification, 2006], [OASIS WS-BrokeredNotification, 2006] and [OASIS WS-Topics, 2006]. These specifications introduce a set of concepts and web-service interfaces of which some are introduced in Section 1.3. Each interaction actor (e.g. consumer or producer) must implement one or more interfaces defined by WS-Notification specifications.

2.1   Base Notification

The most basic interaction involves three (logical) actors: *subscriber*, *notification consumer* and *notification producer* as shown in Figure 1.
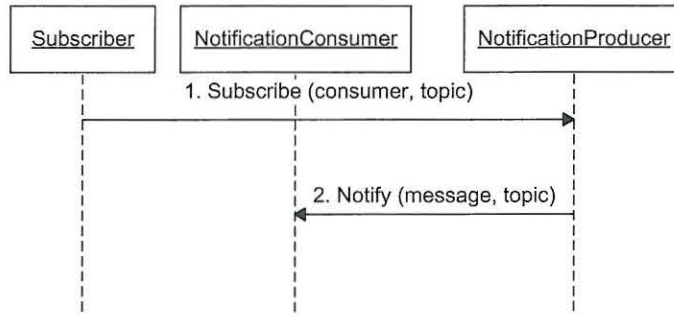
Figure 1   Example of a base notification interaction

In the presented base notification interaction a *subscriber* subscribes a *notification consumer* to a *notification producer* to receive *notifications* of a given *topic*. When a *notification* of a subscribed topic appears as a result of some event on the producer's side, the *notification producer* notifies the *notification consumer*.

A detailed specification for a base *notification consumer* SIP is presented in [NCIA AD 06.05.04.02.G].

2.2   Brokered Notification

The [OASIS WS-BrokeredNotification, 2006] specification introduces the concept of a *notification broker* and a *publisher*. A *notification broker* is a specialized component that is able to receive *subscriptions* and distribute *notifications* from *publishers* that are not necessarily able to receive and manage *subscriptions* by themselves. In the simplest scenario a *publisher* only sends *notifications* to a *broker* which is responsible for dispatching them to all subscribed *consumers* as shown in Figure 2.
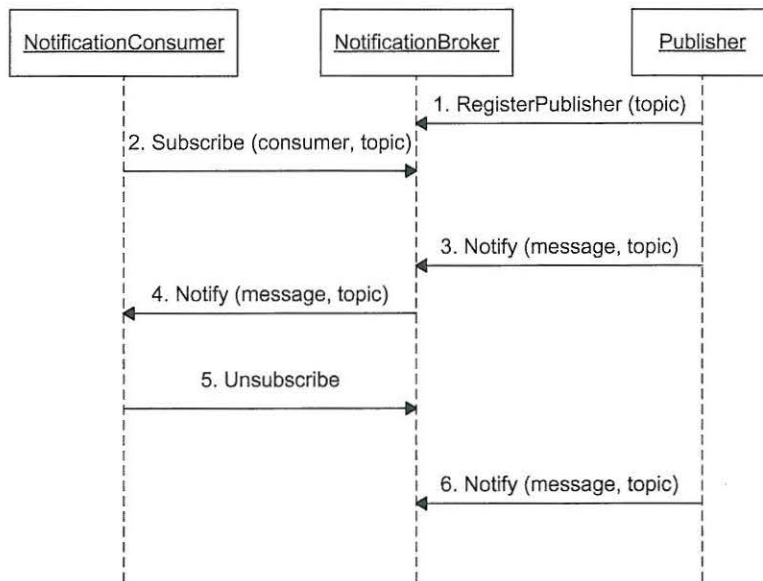

Figure 2   Example of a brokered notification interaction

In the presented brokered notification example a *publisher* producing *notifications* on a specific *topic* registers itself to a *notification broker* capable of providing messages on that *topic*. If the *notification broker* receives a *subscription* for this *topic*, it will forward any subsequent notifications on that *topic* to interested consumers until their corresponding *subscription* expires. If the *notification consumer* is unsubscribed from the *topic* before the *subscription* expires naturally, the *notification broker* should not forward any further notifications.

A *notification broker* acts as *notification consumer* in its interaction with a *publisher*, and acts as a *notification producer* in its interaction with an actual *notification consumer*. For more complex interactions possible between a *notification broker* and a *publisher* please see Section 4 of [OASIS WS-BrokeredNotification, 2006].

In the simplest scenario a *notification consumer* implementation is only able to receive *notifications*. *Subscriptions* are made on its behalf by a *subscriber* implemented in a separate component as shown in Figure 1. For example, a specialized *subscriber* component may create, suspend, renew and cancel *subscriptions* on behalf of multiple *consumers*. In other scenarios a given implementation of a *notification consumer* can also implement the logic of a *subscriber* and subscribe/unsubscribe itself to *producers* or *notification brokers* as presented in Figure 2.

A detailed specification for a basic *notification broker* with *subscription manager* SIP is presented in [NCIA TR/2012/SPW008000/24, 2012].

2.3   Notifications

Notifications can be sent in two forms: as "raw" and unwrapped application-specific messages and as application-specific content wrapped in a "Notify" message as described in Section 3.2 of [OASIS WS-BaseNotification, 2006].

The *notification* message is defined as "one-way only" (see Section 3.2 of [OASIS WS-BaseNotification, 2006]), and without receiving acknowledgements the *notification producers* will not be able to verify that the *notifications* are correctly received by *notification consumers*.

3   RELATED SIPS

This SIP for Publish/Subscribe Services is the overarching parent document for the following two SIPs, which profile the interfaces of the main components in a typical Publish/Subsribe scenario:

- *"Service Interface Profile for a Publish/Subscribe Notification Broker with Subscription Manager"*, described in [NCIA AD 06.05.04.02.F]
- *"Service Interface Profile for a Publish/Subscribe Notification Consumer"*, described in [NCIA AD 06.05.04.02.G]

4   REFERENCES

[IETF RFC 2119, 1997]:
Internet Engineering Task Force Request for Comments 2119, "Key words for use in RFCs to Indicate Requirement Levels", S. Bradner, IETF, Sterling, Virginia, US, March 1997.

[NAC AC/322(SC/1)N(2009)0015 (INV), 2009]:
North Atlantic Council / Euro-Atlantic Partnership Council Notice EAPC(AC/322-SC/1)N(2009)0015 (INV) , "Core Enterprise Services Framework v1.2", NAC, Brussels, Belgium, 30 Apr 2009 (NATO/EAPC Unclassified).

[NC3A RD-2814, 2009]:
NATO Consultation, Command and Control Agency Reference Document 2814, "Bi-SC AIS/NNEC SOA Implementation Guidance" (provisional title), J. Busch, S. Brown, R. Fiske, G. Hallingstad, M. Lehman, NC3A, The Hague, Netherlands, unpublished draft, June 2009 (NATO Unclassified).

[NCIA AD 06.05.04.02.F]:
NATO Communications and Information Agency, Agency Directive 06.05.04.02.F, "Service Interface Profile for a Publish/Subscribe Notification Broker with Subscription Manager", V. de Sortis, M. Lehmann, R. Fiske, L. Schenkels, D. Gujral, NCIA, The Hague, Netherlands, to be published in 2014, (NATO Unclassified)

[NCIA AD 06.05.04.02.G]:
NATO Communications and Information Agency, Agency Directive 06.05.04.02.G, "Service Interface Profile for a Publish/Subscribe Notification Consumer", V. de Sortis, M. Lehmann, R. Fiske, L. Schenkels, D. Gujral, NCIA, The Hague, Netherlands, to be published in 2014, (NATO Unclassified)

 [NCIA TR/2012/SPW008000/30, in prep.]:
NATO Communications and Information Agency Technical Report 2012/SPW008000/30, "Messaging Service Interface Profile Proposal" *(provisional title)*, R. Fiske, M. Lehmann, NCIA, The Hague, Netherlands, in preparation (NATO Unclassified).

[OASIS WS-BaseNotification, 2006]:
Organization for the Advancement of Structured Information Standards (on-line), http://www.oasis-open.org, Web Services Base Notification 1.3 (WS-BaseNotification), at http://docs.oasis-open.org/wsn/wsn-ws_base_notification-1.3-spec-os.pdf, 1 October 2006, viewed 30 March 2011.

[OASIS WS-BrokeredNotification, 2006]:
Organization for the Advancement of Structured Information Standards (on-line), http://www.oasis-open.org, Web Services Brokered Notification 1.3 (WS- BrokeredNotification), at http://docs.oasis-open.org/wsn/wsn-ws_brokered_notification-1.3-spec-os.pdf, 1 October 2006, viewed 30 March 2011.

[OASIS WS-Topics, 2006]:
Organization for the Advancement of Structured Information Standards (on-line), http://www.oasis-open.org, Web Services Topics 1.3 (WS-Topics), at http://docs.oasis-open.org/wsn/wsn-ws_topics-1.3-spec-os.pdf, 1 October 2006, viewed 30 March 2011.

[OASIS WS-ReliableMessaging, 2007]:
Organization for the Advancement of Structured Information Standards (on-line), http://www.oasis-open.org, Web Services Reliable Messaging (WS-1 ReliableMessaging) Version 1.1, at http://docs.oasis-open.org/ws-rx/wsrm/v1.1/wsrm.pdf, 7 January 2008, viewed 21 July 2011.

5   ABBREVIATIONS

Bi-SC   Bi-Strategic Command

CES   Core Enterprise Services

IETF   Internet Engineering Task Force

NNEC   NATO Network Enabled Capability

OASIS   Organization for the Advancement of Structured Information Standards

RFC   Request for comment

SIP   Service Interface Profile
SOAP   Simple object access protocol

XML   Extensible markup language