

RosettaNet Implementation Framework (RNIF) 2.0

Benefits of RNIF 2.0 and Comparison to RNIF 1.1 February 2000

What is RosettaNet?

Launched in June 1998, RosettaNet is an independent, self-funded, non-profit consortium of major Information Technology (IT), Electronic Components (EC) and Semiconductor Manufacturing (SM) vendors. Dedicated to the development and deployment of open electronic commerce standards that align the business processes between partners in high tech supply chain, RosettaNet specifications allow manufacturers, distributors, resellers and end-users to leverage the Internet to exchange business documents across the entire supply chain, on a global basis.

A pioneer to the field of XML based open business-to-business electronic commerce standards, RosettaNet is the first industry consortium to deliver an exhaustive set of standards that encompasses the entire high tech supply chain spectrum. In addition to specifying a comprehensive set of XML based standard business document schemas and data dictionaries, RosettaNet normalizes and specifies the common business processes between trading partners (known as PIPs), and an open and interoperable protocol for the networked applications that execute the business processes.

RosettaNet Standard Tracks

RosettaNet standardization efforts can be divided into three broad groups of data format, business process and protocol specifications:

RosettaNet Dictionaries

The RosettaNet dictionaries define common set of properties for use by the business process (PIP) specifications and associated business documents and

guidelines. RosettaNet dictionaries are classified as Business and Technical Dictionaries.

The *RosettaNet Business Dictionary* defines the properties used in basic business activities between trading partners. These are the Business Properties (e.g. business address), Business Data Entities (e.g. ActionIdentity) and Fundamental Business Data Entities (e.g. BusinessTaxIdentifier, AccountNumber) etc. There is only one business dictionary that encompasses all supply chains (EC, IT etc).

The *RosettaNet Technical Dictionary* defines properties for products, components/devices and services that span the EC, IT and other supply chains. Note: formerly distinct *EC Technical Dictionary* and the *IT Technical Dictionary* are now integrated into the *RosettaNet Technical Dictionary*.

RosettaNet PIPs

RosettaNet Partner Interface Processes™ (PIPs™) define the specific sequence of steps required to execute a business processes between supply-chain partners. This is also known as the business process choreography. Examples of such processes include purchase order management and distribution of new product information.

In addition to the business process choreography, a RosettaNet PIP specification includes the specification of structure and content format of the business documents exchanged (XML DTDs and guidelines) and the time, security, authentication and performance constraints on these interactions.

Specification of the PIPs is limited to publicly visible interactions and interfaces between the trading partners. Hence these are also referred to as the public processes. In contrast, private processes are business processes within an enterprise that trigger the execution or are triggered by the execution of the public processes or the PIPs.

RosettaNet Implementation Framework

The RosettaNet Implementation Framework or RNIF for short provides the fundamental plumbing required to execute business processes between the trading

partners, in an open, interoperable, secure and platform & implementation independent way.

The focus of this paper is RNIF, more specifically RNIF version 2.0, a recently approved successor to the RNIF 1.1 specification.

What is RNIF?

At a high level, RNIF defines the overall RosettaNet business message format for exchange of the business documents, with elements to support authentication, authorization, encryption and non-repudiation; details of the bindings for the transfer protocols (e.g. HTTP); and the specification for a reliable exchange of messages between partners.

Purpose of RNIF

The PIP specifications define the document exchange choreography and the XML schemas for the individual business documents involved. The format of these schemas varies on per PIP and the specific business document type basis, based on the underlying business purpose that the document serves. Hence it is necessary to define an overall envelope/container format that stays constant and consistent for all exchanges within which all business documents are exchanged as payload. RNIF specifies such an envelope format that is also independent of the specific transfer protocol used to transmit the message between partner nodes.

It is necessary to capture the context information for the specific step in PIP process that the payload business document executes and the attributes of the payload document(s), in a PIP and business document independent way. RNIF specifies the XML schema for a header document called the Service-Header for this purpose, an instance of which must always precede a business document instance, in a RosettaNet message.

RNIF specifies and provides for a consistent mechanism to digitally sign and or encrypt all RosettaNet messages (as needed), independent of the transfer protocol, PIP and the specific business document being exchanged.

RNIF specifies a reliable messaging mechanism based on Acknowledgements and supplies a set of standard choreography models that all PIPs must follow.

RNIF owns and specifies the schemas for the Acknowledgement and Exception (error) messages.

RNIF specifies the transfer protocol level bindings for all the supported transports so that RosettaNet messages are exchanged in a consistent and interoperable way.

The section below describes the core features of RosettaNet Implementation Framework version 2.0 and the subsequent section describes the key differences between versions 1.1 and 2.0 of the specification.

Core RNIF 2.0

At the heart of the RNIF 2.0 core specification is the RosettaNet Business Message, a transfer protocol independent container that packs together the business payload, associated header components and other entities like the optional digital signature that must all be exchanged as a unit between two end-points of a RosettaNet interaction.

RNIF 2.0 specifies the use of MIME multipart/related type for the basic enveloping construct to pack together different elements of a RosettaNet Business Message and the S/MIME v2 multipart/signed and the application/pkcs7-mime enveloped-data types for digitally signing and content enveloping purposes respectively.

Figure below shows the components of a basic RosettaNet message encased in a multipart/related envelope:

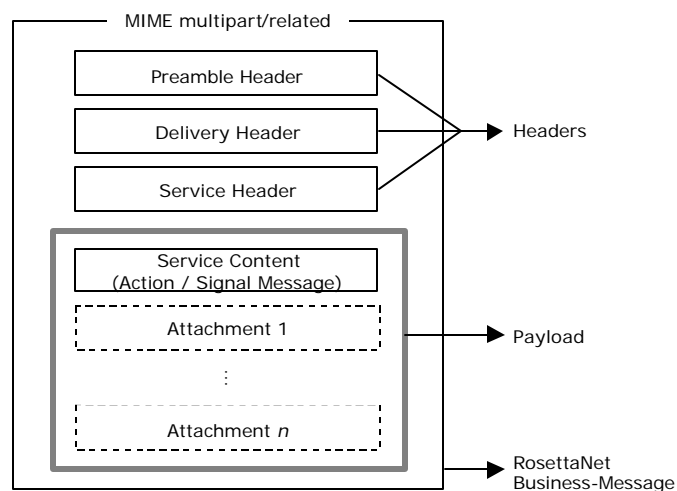


Figure 1 Basic RosettaNet Business Message Components

In addition to the payload, all RosettaNet Business Messages contain an instance of the Preamble, Delivery and Service Header components, each of which are separate XML document instances, schemas for which are also defined by the RNIF 2.0 specification. Preamble and Service Headers are modified and optimized version of the RNIF 1.1 equivalents. However the Delivery header component is new in version 2.0, and contains elements to support the transmission of the RosettaNet messages through third party intermediaries such as hub providers.

As shown, the payload part of an RNIF 2.0 RosettaNet business message provides support for optional attachments.

A transfer protocol independent RosettaNet business message MIME or S/MIME packaged as described above is transmitted between two RosettaNet endpoints. RNIF 2.0 supports HTTP(S) and SMTP protocols with provision for future addition of other transfer protocols like FTP.

The figure below shows the network application protocol stack at a RosettaNet end-point as it exchanges a RosettaNet Business Message.

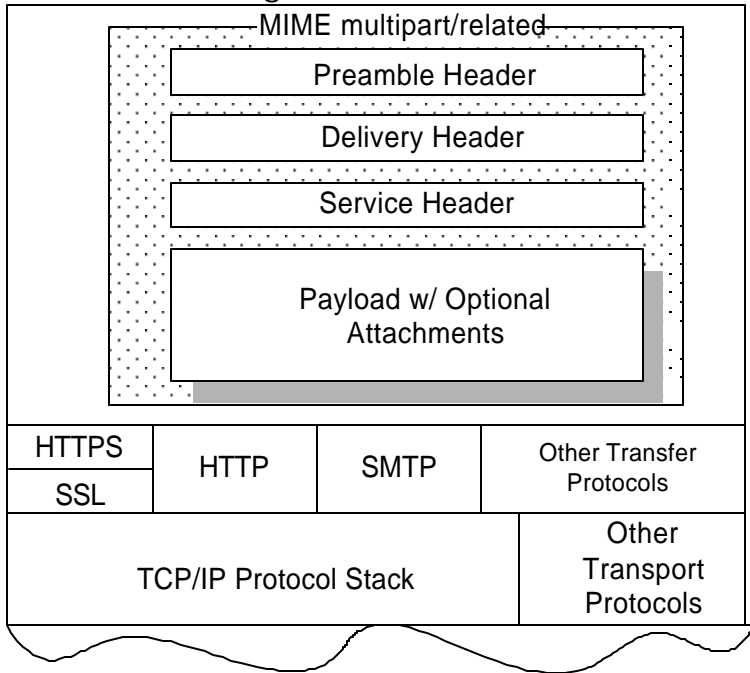


Figure 2 RosettaNet Application Protocol Stack View

The core specification promotes interoperability of two independent implementations of RosettaNet by providing a clear specification of transfer level parameters for all supported

protocols and the associated secure transfer protocols, such as SSL V.3 with the HTTP protocol.

The RNIF 2.0 core specification owns and defines the schemas for the Acknowledgement and Exception messages (known as the *signal* messages) that govern the reliable messaging aspects of the PIP message exchanges. The core specification also defines the standard message exchange patterns with retry and timeout constraints and the corresponding interaction flow diagrams for one-action and two-action activity PIPs for both synchronous and asynchronous modes of interactions. The interaction flows clearly identify the logical sequence of steps involved in processing a RosettaNet message such as, message structure validation, grammar/schema validation etc., and the error or exceptions conditions that could occur at different processing steps.

The core specification includes the authentication, authorization, encryption and non-repudiation requirements essential for conducting secure electronic business over the Internet.

Authentication is the process of reliably establishing the identity of the party that is communicating. RNIF requires the use of digital signatures that conform to the S/MIME v.2 specification and the associated digital certificates issued by a mutually trusted third party, as an effective way to establish the authenticity of the communicator.

Authorization is the process of making sure that a sending party or sending party's organization is permitted to send the subject message (or perform the subject business action). In addition to the identity as specified in the service header, RNIF recommends the use of digital signatures that conform to the S/MIME v.2 specification and the associated digital certificates to establish the identity of the communicating party for authorization purposes also.

Non-Repudiation is the process of assuring that a particular person or service sent or received a message. For example, a sender would be unable to deny having sent a message called non-repudiation of origin & content; and a receiver would be unable to deny having received a message called non-repudiation of receipt. Again digital signatures on messages sent are used to provide both non-repudiation of message origin & content and non-repudiation of receipt; the latter with a signed receipt acknowledgement.

Use of digital signatures also aids in detection of tampering or in transit corruption of messages.

Key Differences Between RNIF 2.0 and RNIF 1.1

RNIF 2.0 is a much more robust, well-defined, comprehensive and interoperable specification when compared to RNIF 1.1. Fixes several deficiencies and inconsistencies in version 1.1 of the specification, and adds richer and needed functionality. Uses well-established standards where available, and eliminates proprietary aspects of version 1.1 specification. Gives clear specification of all ambiguous areas identified during the field trial and production use of the RNIF 1.1 specification.

Transport independent Message Format

RNIF 2.0 defines a transfer protocol independent RosettaNet Business Message format that is based on MIME and S/MIME standards. This format replaces the RosettaNet Object (RNO) format in version 1.1, which was a proprietary format adapted from OBI (Open Buying on Internet).

Multiple Transfer Protocols

RNIF 1.1 defined support for HTTP(S) transfer protocol only. RNIF 2.0 adds support for SMTP transport protocol, in addition to HTTP/HTTPS with a design based on the transfer protocol independent message format, which can support other transfer protocols such FTP, message queues in the future.

Transfer Protocol Level Debug Headers

RNIF 2.0 defines a mechanism to use transfer protocol level debug headers. The debug headers carry basic information on the message being transferred between two nodes so that, meaningful errors can be returned to the sender even when the received message can not be processed at all. This can be useful when two independent solutions just begin to interoperate when even the headers components may not be parsed successfully or even if the message cannot be unpacked successfully.

This facility did not exist in version 1.1 and is new for version 2.0 of RNIF.

Encryption of Payload and Service Header

RNIF 2.0 defines an S/MIME based mechanism to encrypt the payload part of a RosettaNet Business Message with an option to encrypt the Service Header also if needed. Encryption of payload ensures privacy and prevents unauthorized access to payload during transit. Even the service header carries sensitive information such as the name of the PIP, subject business action type etc., which can be used derive meaningful statistical information, regarding business activities between trading partners. Hence RNIF 2.0 provides an option to encrypt the service header also if desired.

RNIF 1.1 did not define support for encryption facilities.

Clean-up of Service Header and Manifest

The RNIF 1.1 version of the service header has several inconsistencies and improper nesting of ProcessControl, ServiceRoute, Role, TransactionControl and ActionControl sub-sections. RNIF 2.0 fixes all the inconsistencies, properly nests all the sub-sections and adds a Manifest element to properly catalog the contents of the payload part of the RosettaNet Messages, in the header.

Digital Signatures

RNIF 1.1 defined a proprietary mechanism to add detached digital signatures to RosettaNet messages using the RNO format. RNIF 2.0 replaces this proprietary mechanism with an S/MIME based one for digitally signing RosettaNet Business Messages.

Support for Attachments

RNIF 2.0 adds formal support to add attachment to RosettaNet Service Content and a mechanism to reference to the attachments from the Service Content. RNIF version 1.1 does not have formal support for attachments, even though a mechanism similar to the one defined by version 2.0 was recommended in the form a Technical Recommendation.

Enable Hub Based Routing

RNIF 2.0 adds a new header element called Delivery Header to the RosettaNet Business Message, which has minimal set of attributes needed to route a message from the sender to the eventual recipient, even when the service header is encrypted. This, in conjunction with the unique *messageTrackingId* field of

the delivery header facilitates the routing of the message through third-party intermediaries such as hub providers.

Support for Registry Driven Requests

RNIF 2.0 adds support for registry driven requests, where sender of a business message is not a known party with pre-established configuration for reply addresses and a partner-identification or digital certificates, as with the known partners. This is useful when a party typically wants to permit look up of publicly available information such as specifications for a product etc., by unknown parties. *UnknownInitiatingPartner* field of the RNIF 2.0 Service Header is specifically designed for this purpose.

Third-party Business Content

RNIF 2.0 adds support for shipping third-party business content in RosettaNet Business Messages, as sanctioned (in future) by RosettaNet PIPs. For example, a purchase order request business message defined by other consortiums (e.g. xCBL) can be exchanged in PIP 3A4 if sanctioned by RosettaNet.

Synchronous Transactions

RNIF 2.0 defines support for synchronously exchanging request and response business messages on the same HTTP connection. This is required by PIPs such as 2A9 that require time sensitive responses from the responding party.

Refinement of Reliable Messaging

RNIF 2.0 refines and simplifies the reliable messaging aspects, by eliminating activity level retries and discontinuing the use of the AcceptanceAcknowledgement signal message. It should be noted that none of the PIPs defined thus far by RosettaNet had actually used the AcceptanceAcknowledgement signal message.

Changes that impact the PIP specifications

IFV integration into RNIF: RNIF 2.0 integrates the Implementation Framework View part of the PIP specification into the RNIF, as an appendix that can be used to derive the IFV parameters in a straightforward and consistent fashion for all PIPs. Hence all future versions of the PIP specifications will not contain the IFV section.

Elimination of the Agent based Dialogs: RNIF 2.0 calls for elimination of Agent based dialogs from the PIP specifications. Agent based dialogs have been the most confusing and incomplete and inconsistently specified parts of the PIP specifications. RNIF 2.0 adds an explanatory appendix on the need for and the significance of Agent based dialogs and recommends that Agent interactions should not to be standardized, as they are tightly coupled with the end-user interfaces (UIs) that tend to be very implementation and solution specific.

Interoperability of RNIF 1.1 and 2.0

Due to the changes in the RosettaNet Business Message format and differences in the transfer protocol level binding (and other key differences as described above), RNIF 1.1 and RNIF 2.0 are not mutually interoperable.

Summary

RosettaNet Implementation Framework 1.1 provided the first usable version of a specification that partners could build solutions for and interoperate by exchanging the RosettaNet PIPs. However, use of RNIF 1.1 based solutions by the partners has shown that RNIF 1.1 has several deficiencies and inconsistencies that needed to be fixed. The RNIF version 2.0 specification fixes these deficiencies and inconsistencies. It is a much improved, robust, comprehensive and a far superior specification when compared to its predecessor 1.1 version that adds much needed newer functionality such as support for multiple transfer protocols, support for hub based routing, support for attachments, encryption of the payload etc., in addition to aligning with industry standards where applicable there by eliminating all proprietary aspects of RNIF 1.1.

RNIF 2.0 specification is currently undergoing validation by a number of solution providers. RosettaNet has stated that RNIF 1.1 will continue to be supported until next major revision of the RNIF specification. However, you should check with RosettaNet for the current support plan.

Additional information on RosettaNet Implementation Framework versions 1.1 and 2.0 and all other RosettaNet specifications can be found on the RosettaNet website at <http://www.RosettaNet.org>.