



doc. nr.	ISO/IEC JTC1/SGFS N 940	
date	1993-06-10	total pages
item nr.	supersedes document	

Secretariat: Nederlands Normalisatie-instituut (NNI)  
Kalfjeslaan 2 P.O. box 5059  
2600 GB Delft  
Netherlands

telephone: + 31 15 690 390  
telefax: + 31 15 690 190  
telex: 38144 nni nl  
telegrams: Normalisatie Delft

Title: ISO/IEC JTC1/SGFS  
ISO/IEC JTC1 Special Group on  
Functional Standardization

Secretariat: NNI (Netherlands)

Title : AFNOR contribution on the subject of PAS (Publicly Available Specifications) and especially APIs (Application Programmatic Interfaces)

Source : AFNOR

Status : For discussion during the SGFS Plenary Meeting, July 5-9, 1993, Seoul, Korea

Note :



**association  
française  
de normalisation**

Tour Europe - Cedex 7  
92049 Paris La Défense

Tél. : 33 (1) 42 91 55 55  
Télex : AFNOR 611 974 F  
Télécopie : (1) 42 91 56 56  
Minitel : 3616 AFNOR

**AFNOR contribution on the subject of PAS (Publicly Available Specifications) and especially APIs (Application Programmatic Interfaces).****0- References (documents appended) :**

OIW contribution EWOS/TLG/93/054  
EWOS TA Special Session Report : EWOS/TA/93/079, SGFS/OSE N 27  
CN 21 N 1942 (AFNOR response to SC 21 N 7425)

**1- Definition of a PAS**

PAS : Specification which is publicly available (either free or reasonable fee, non-discriminatory access), usually coming from consortia (hopefully committed to work in close cooperation with ISO/IEC e.g. applying for C-liaison status) and recognised as having a market influence.

Several types of specifications are candidates for being PAS :

- . Programming Interfaces (APIs are in this category)
- . Protocols
- . Formats
- . Human Computer Interface

There may be more than one PAS for a given purpose.

**2- Need for referencing/using PAS**

Public procurement, and many other users have expressed the need of "using" PAS in their procurement specifications. They have asked the Workshops, and SGFS, to pave the way for allowing this.

EWOS has suggested to have a pragmatic approach and to take TP and NM APIs as trial examples.

There are other activities in other forums, for example some X/OPEN APIs (X400, X500 for instance) are progressed as IEEE standards and should be submitted as fast-track to JTC1 by ANSI.



### 3- The way forward

Several elements for discussion have been mentioned :

1. - How to select the right PAS and what are the selection criteria? One of the selection criteria should be : specification coming from organization with a liaison status with on or several ISO subcommittees or workgroups (e.g. C-liaison).
2. - Should the selected PAS evolve into standards ? The answer should be yes.
3. - Where to "register" a PAS and how?
4. - How is a PAS maintained and what is the process ?
5. - How can we study and register the fact that standards/profiles and PAS can work together (for example, that a API is useable over a given profile).  
And specifically for APIs:
6. - Where architecturally should APIs be defined ?

An user requirement on APIs is that there is urgency to identify, select and possibly adopt existing APIs in a timeframe shorter than the one necessary to formal standardization process.

The OIW contribution on PAS states that "The PAS must neither overlap with nor conflict with an existing formal standard or formal standard under development ...". AFNOR strongly agrees with the intent of this statement, but would like to point out that for example the fact that a NWI for a SPI (Standardized Programmatic Interface) when initiated in JTC1 should not prevent the pragmatic referencing of existing APIs in the short term. It would rather be desirable in such a case to provide a process for revision of such a pragmatic reference when the standard is mature.

The workshops groups on OSE are trying to provide a framework, based on the concept of components, that will include PAS and show their relationships with the other components. It should be contributed to SGFS when mature.

SGFS may have a role to play in this respect. Possible actions include:

- collect, as JTC1 has previously recommended, the user requirements on this topic.
- provide a place where to list (register ?) some PAS.
- mention in an informative-only way, those PAS that may be used in conjunction with the profiles, especially the APIs that are compatible with the profiles, when the information has been provided by NBs or the S-Liaisons and has been harmonized .

This last point seems an important one from a user point of view, and practical experiments are ongoing : for example the three workshops have commented already on the X/OPEN XMP API (for CMIP) and on its adequation to the CMIP profiles (AOM11/AOM12). If the API is modified along with the comments provided, then the workshops will be able to pass to SGFS the information (harmonized) that this API is able to complement the AOM11/12 profiles. It seems a useful information to record. Such information may be available in other domains.



## Publicly Available Specifications

NOTE - This Annex forms an integral part of the OIW Procedures.

Users observe that increasingly there are specifications which provide needed extensions to the international standards, have broad consensus, and can meet user OSE business needs in advance of the completion of the formal standards process. Many users would like to be able to exploit this consensus in their procurement process sooner rather than later.

When proposing the use of a Publicly Available Specification, a SIG makes its case, using the following guidelines. The OIW Plenary would be responsible for accepting or rejecting the SIG's proposal using the voting rules of the OIW.

The Publicly Available Specification must neither overlap with nor conflict with an existing formal standard or formal standard under development. That is, if a formal standard exists or is under development that provides the same function as the proposed Publicly Available Specification, then the Publicly Available Specification may not be introduced as the basis for OIW Implementation Agreements; if a Publicly Available Specification adds functionality then it must be engineered to augment the formal standards in such a way that interoperability among systems implementing the formal standards is not precluded.

A SIG would only propose to reference a Publicly Available Specification in an Implementation Agreement when it provides a technical function that meets a clear and widespread user requirement. The specific reference must be labelled as a "Publicly Available Specification" in the agreement.

In exceptional circumstances driven by user requirements, a SIG may propose to reference a de facto standard in an Implementation Agreement where the de facto standard does functionally overlap an existing formal standard but otherwise meets the criteria for a Publicly Available Specification; this would be strictly limited to cases where the SIG can demonstrate that the agreement will expedite the migration (e.g., facilitating a gateway or interworking) from the de facto standard to a formal standard in multi-vendor environments.

Where more than one Publicly Available Specification might serve as the basis for OIW Implementation Agreements for the same technical function, the SIG proposing to use a Publicly Available Specification will recommend which among the several candidates should be used and why. The OIW Plenary will make the final choice among competing Publicly Available Specifications in response to specific user requirements. Whenever possible only one Publicly Available Specification should be used as the basis of Implementation Agreements for any specific technical function.

In proposing the use of a Publicly Available Specification as the basis of Implementation Agreements, the SIG must document that the specification meets the following criteria:

- a) Common description; the specification should be described using conventions, including conformance statements, appropriate for the existing formal standards which the specification augments. For example, a Publicly Available Specification describing a new networking service and a supporting protocol should be described with a service and protocol specification using the conventions established for OSI standards;
- b) Stability: the specification will not change except as required to fix technical and editorial errors. The OIW must be free to change and amend the specification as required to fix technical and editorial errors and to make it suitable for submission to the formal standards process.
- c) Completeness; the specification must be sufficiently complete so as to allow useful and predictable implementation of the complete functionality from scratch. For example, an interface specification would not qualify if it simply permits standardized access to an otherwise proprietary implementation which provides the functionality.



## Draft Report on the 2nd TA Special Session on Open Systems

A special session of the TA was held from 1400 to 1700 on Wednesday 24th February. It was chaired by the TA Chairman, Dr W Black. The meeting was a continuation of an initial special session on Publicly Available Specifications (PAS) and User Requirements held during TA19.

The objective of the meeting was to finalise the definition the EWOS position on PAS and User Requirements and to agree actions consequent on this position.

This report deals in turn with the discussions on User Requirements and Publicly Available Specifications. It closes with a summary of the final conclusions and actions agreed in both of these areas.

### 1. User Requirements

- 1.1 Introductory presentations were given on the subject of user requirements by Mr N Hopkins (See EWOS/TA/93/072) and Mr L Bertuzzi (see EWOS/TA/93/04). These are briefly summarised below.

Mr Hopkins presented the thoughts of the CCTA on the further development of Open Systems. He stressed the growing customer demand to be able to easily translate their total business requirements into comprehensive packages - profiles - of "base" interface specifications which can be used to build the IT function the business requires.

Mr Hopkins highlighted the "gap" which he believes currently prevents the achievement of this translation, namely the lack of a language to enable business requirements to be translated into corresponding technical specifications.

He suggested that, in order to bridge this gap, an intermediate level of profiles is required which package base specifications (OSI protocols, OSI profiles and APIs) into building blocks of IT function which meet identified user needs and which can be provided by procurable products. These building blocks would provide the link between the language of user requirements and that of technical specifications.

Mr Hopkins expressed the desire of CCTA to work with EWOS to produce such harmonised building blocks as ISPs and to develop a methodology for using them to produce complete system specifications. He said that this will also require that a means be found to allow the use of PAS in such profiles. He also stressed the need for the plan for these activities to be made public so that work is not duplicated.

Mr Tucker commented that EG OSE supported this approach and that the group have been working on the definition of these building blocks and how they can be related to user language. They have so far had discussions with EG NM, SEC, DBE and TP to work out what specific building blocks might be developed in these functional areas. TP and NM have now announced two new work items based on the idea of such building blocks. He said that these might be used as test cases in this approach.



Mr Bertuzzi emphasised the need for some form of presentation gateway to ensure that the technical output of EG-OSE reflects user requirements. He believed that the initiative should come from the National Bodies in the JTC1 forum. He referred to the document "Framework for the Registration of Requirements for the OSE" in which Mr G Cunningham outlines several points concerning the collation and definition of user OSE requirements at an international level.

## 1.2 The meeting then discussed possible mechanisms for collating User Requirements.

There were several suggestions that a new forum was needed to bridge the activities of the workshop and the user community. Several participants suggested that OSITOP, via the X/Open User Council, could be explored by EWOS as such a forum, thereby also ensuring that requirements work is not duplicated. It was also suggested that the TA should charge the EGs to initiate this relationship.

Mr Tucker recommended that such fora as OSITOP and the X/Open User Council should be used to identify priorities rather than as a means of specifying requirements. He suggested that the building block approach could be used as a way of defining the problem. Provisional building blocks could be developed by EWOS on the basis of its existing knowledge of requirements and could then be adapted according to user responses to them.

The Chairman summarised the view of several participants that the mechanism already exists in the form of EWOS workshops for users to provide input. OSITOP is able to attend EG and TA meetings. There is therefore no need to institute a separate mechanism for User Requirements. It is the role of the EGs to liaise with the relevant user groups. He added that, as a conclusion from this, the TA does not need to be involved in establishing a separate User Requirements mechanism.

It was however agreed by the meeting that there is a need to publicise what EWOS is doing in the area of User Requirements and that this should be done at the TA level.

The Chairman further summarised that EWOS, rather than try to define User Requirements itself, should be responsive to those produced by other sources/channels. It therefore needs to concentrate on who can supply requirements input and what EWOS can provide to help them do so.

The meeting agreed that the Steering Committee should play a role in channelling higher level business needs to the workshop.

Finally, Mr Tucker reaffirmed that EG OSE are willing to act as the channel for and custodian of User Requirement input.

## 2. Publicly Available Specifications

Mr Wakker highlighted the need to distinguish between "Documentation" profiles and "Implementation" profiles (see EWOS/EG-OSE/93/27). In the former, gaps in the standards infrastructure may be identified by simply describing the functionality for which there is no standard. In the latter however, a non-standard conforming product may need to be specified to complete the profile in order that it may be implemented. Since SGFS regard ISPs as *documenting* function only and therefore are unlikely to accept the inclusion of PAS, two types of profiles will need to coexist: official OSE profiles which identify gaps and "Implementation" profiles which identify PAS to fill gaps.

In response to Mr Wakker's comments the Chairman raised the question as to whether the Regional Workshops should, given that JTC1 will be slow in responding to their need to



incorporate PAS in ISPs, proceed in producing their own "implementation" documents.

Ms Valet felt that EWOS should not wait for SGFS fasttrack. Mr Wood said that EWOS should wait to understand SGFS' view on PAS for APIs in order to establish some structure for dealing with them. Many members felt that SGFS and the National Bodies should be kept informed of the work EWOS is doing in this area but that this should not be allowed to slow down this work.

Mr Tucker said that, because the range of PAS is so diverse, a general approach could not be defined. He proposed that EWOS deal only with those PAS of immediate interest to it, for instance X/Open APIs, and proceed with one or two examples as test cases. He suggested that those APIs planned to be incorporated in either the new TP (EWOS/TA/93/14) or NM NWIs could be used as guinea-pigs. The meeting endorsed this approach.

Following from this a question was raised as to the procedure EWOS should adopt for dealing with PAS for APIs.

It was agreed that, for each API being considered, a paper should be prepared for input to SGFS explaining what EWOS plans in the case of that API.

As an immediate next step, such a paper will be jointly prepared by EG OSE and EG TP on the TP NWI which will incorporate an API. This paper will be presented at the next Special Session of the TA and submitted for approval by the TA as input to SGFS.

### 3. Summary

#### 3.1 User Requirements

The mechanism already exists within the EWOS framework for users to provide input to OSE profile definition. It is the responsibility of individual EGS to form a liaison with the relevant user groups. There is no need for the TA to be involved in defining a separate mechanism.

Specific level requirements should be fed to the EGs and general level requirements to EG OSE.

EG OSE will be the custodian of User Requirements in this field.

EWOS should publicise its activities in this field, probably via the TA.

The SC has a role in liaising between EWOS and users at the higher level of business need.

#### 3.2 PAS

EWOS should proceed with the incorporation of PAS in profiles.

Rather than define a general approach to the issue of PAS, EWOS should deal with each PAS (initially X/Open APIs) as the requirement for it arises.

SGFS should be kept informed, by means of a TA approved paper, of what EWOS' plans are concerning each PAS it has decided to incorporate into profiles.

The new TP NWI (EWOS/TA/93/14) will be adopted as a test case. EG TP and EG OSE will jointly present a paper to the next TA special session and for approval by TA21 for submission to SGFS.





**association  
française  
de normalisation**

Tour Europe - Cedex 7  
92049 Paris La Défense

Tél. : 33 (1) 42 91 55 55  
Télex : AFNOR 611 974 F  
Télécopie : (1) 42 91 56 56  
Minitel : 3616 AFNOR

Ligne directe :

**CNTI/CN 21 F 1942**

**TITLE: COMMENTS ON THE DRAFT REPORT ON PROGRAMMATIC  
INTERFACES**

**SOURCE: AFNOR**

**REFERENCE: SC21 N7425**

**STATUS: APPROVED AFNOR COMMENTS**

**Proposal :**

Replace 1st paragraph with: "Current work on API specification has made available considerable experience and feedback on what may be considered good practice in the area. ~~The existing (non-standardized) APIs are providing building blocks widely used in the industry.~~ In certain areas, they have permitted to provide stable development environments and provide for portability of applications between different vendors hardware and software. Yet areas for improvement have been identified."

Move last paragraph to become 2nd, and change it to read: "Many organizations are actively developing interface specifications, but the work should be complemented with an overall plan. The problem for users is to select APIs that will complement one another to accomplish the user's objectives. For example, a user who needs a transaction-oriented RPC mechanism needs some guidance for the selection of the appropriate API."

**2.3 PAGES 3,4 CLAUSE 3****Rationale:**

Simple interfaces can be adequately specified using natural language comments. ~~Object oriented languages are not required for all types of interfaces:~~ some of the existing (non-standardized) API specifications using non object oriented languages may well continue to exist over time as useful building blocks of more elaborate (future) implementations. Note also that no International Standards for object-oriented programming languages are available yet.

**Proposal:**

In 2nd paragraph, change the last part of the sentence (after ;) to read:

"the nature of the functions performed by the interface has to be supplied in natural language comments. While this is perfectly adequate for simple interfaces (i.e. most of the interfaces defined today), the specification task becomes difficult when the complexity of the interface increases".

Delete 1st sentence of last-but one paragraph ("The language-specific ... behavioral dependencies at the interface"). Join the rest of the paragraph with the text proposed above and finishing with "the specification task becomes difficult when the complexity of the interface increases".

In 3rd paragraph, change from 3rd sentence ("Bodies, such as X/Open ..) until the end to read: "Where such a requirement is identified, SPI specifications could be programming language independent and cover thus object programming or non-object programming. This could reduce conversion efforts and ensure that instantiations of an interface in different languages have consistent semantics".

**2.4 PAGES 3,4 CLAUSE 3****Rationale:**

The example quoted, starting with "Consider the following operation signature ..." and the paragraph that follows are a good illustration of the advantages brought by the ODP framework over "old-fashioned" specifications. Yet this clause is supposed to look at advantages and problems of current practice; this example should be under a different heading.

**Proposal:**

Create a sub-clause 3.1, with title "Advantages of a detailed reference model over current practice".

Move text starting on page 3 with "Consider the following example" until end of paragraph "stringent conformance requirements" on page 4 in sub-clause 3.1.

Add an introductory sentence: "The following example illustrates the advantages a detailed Reference Model can bring over current practice."



**2.5 PAGE 5 CLAUSE 4.1 PARAGRAPH 7****Rationale:**

The expression "standard programming interface", although better than "application", is incorrect as it leads to the direct incorporation of interfaces created by any so-called "standard" body. The terminology used in similar circumstances, e.g. JTC1/SGFS, is "standardized".

**Proposal:**

Replace "standard" by "standardized" in the definition of SPI.

**2.6 PAGE 6 CLAUSE 4.2.2****Rationale:**

The discussion on conformance is probably incorrect, as it refers to considerations which fall in the area of relationship between standards rather than of relationship of a product with a standard. What can be considered in terms of conformance is, given an SPI standard, what can be tested of a product. It has to execute certain actions on the interface in response to actions executed on the same interface by the partner on the other side of the interface. These actions must be executed in a syntactically correct way. The actions executed by the partner must be accepted if they are themselves syntactically correct. Errors specified by the standard must be detected and reported as specified. It may be that this is what 4.2.2 wants to say.

AFNOR believes that conformance should not be expressed with respect to a standard of the language independent type. Products are made to work with one language. Therefore conformance should be expressed and tested with respect to language bound SPIs.

**Proposal:**

- Rename clause 4.2.2 "Relationships between standards"
- Delete "conformance" in last line of paragraph 1 ("conformance framework" becomes "framework");
- Remove 3rd bulleted item (and the "and" at the end of bullet item 2).
- Add after bulleted list:

"In particular, for programmatic reference points, the abstract (language-independent) specification of an interface that serves as a reference point, are in general complemented by language binding specifications that define rules for writing language dependent specifications for client interfaces, and a translation mechanism for such specifications that is compliant with the abstract interface specification".

- Start a new clause or sub-clause with title "Conformance framework"

" A conformance framework is needed, expliciting, given an SPI standard, what can be tested of a product.

It should explicit that the product has to execute certain actions on the interface in response to actions executed on the same interface by the partner on the other side of the interface. These actions must be executed in a syntactically correct way. The actions executed by the partner must be accepted if they are themselves syntactically correct. Errors specified by the standard must be detected and reported as specified.

This framework should remain simple, be provided quickly and take advantage of the existing practice in the area of OSI and language standards.

The compliance of an SPI standard to:

- the Framework for SPIs,

- 7 -

Add a paragraph stating that until architectural framework and style guides are available, work on SPIs must nevertheless progress.

#### 2.11 PAGE 8 CLAUSE 6 PARAGRAPH 1 LINE 2

##### Rationale:

The expression "system interfaces" appears, although it has never been mentioned in the preceding text. It is probably a typo for "standardized programming".

##### Proposal:

Correct as needed.

#### 2.12 PAGE 10 CLAUSE 6 PARAGRAPHS 6 AND 7

##### Rationale:

It does not seem to be appropriate in this document to make general assumptions on the style of interfaces that ODP will select. If ODP were to limit itself to RPC, it would suffer from highly restricted capabilities that would unduly limit its value. But such a discussion is for ODP, not for SPIs.

##### Proposal:

Remove the two paragraphs on RPC.

#### 2.13 PAGE 11 CLAUSE 7.1 PARAGRAPH 1

##### Rationale:

How is the decision made that a standard "requires" an SPI? By a NP ballot process?

It is most probable that SPIs to other things than base standards will be required. Actually, this would be preferable in many cases. The new application layer structure ISO 9545 provides a convenient way of placing interfaces above groups of base standards, above ASOs.

It is very likely that SPIs will be needed in conjunction with profiles as defined in JTC1/SGFS.

Conversely, the implications of the work on profiling in the application layer should be considered. When a profile makes restrictions to the use of an application element or object, should this be reflected in the SPIs? And, if so, who is responsible for this work?

Another question is that of which of the N standards which share an interface should be responsible for the definition of the interface. Or should it be a common work? If yes, how is this to be achieved?

##### Proposal:

Expand the concept of "base standards" in 7.1

#### 2.14 PAGE 12 CLAUSE 8.1

##### Rationale:

Clause 8.1 should be reviewed and the first sentence of 8.1 should be amended. In an ideal world (with infinite resources and no time constraints) this might be true, but for the sake of being pragmatic we must rephrase 8.1 in a more short-term perspective.

##### Proposal:

8.1 Proposals for advancing work on SPIs in the short term:

— Style guide for the definition of SPIs.

This is an urgent piece of work for which there are contributions already, some of them coming from SC21/WG7.



FTM is a typo for JTM

It is unclear why only connectionless session and presentation are listed. It would be more appropriate to list the more common mode of operation.

A list of interfaces to functional standards should be included.

#### 2.18 NEW ANNEX D: List of Standardized SPLs

Rationale :

Build from existing experience in ISO/IEC and CCITT.

Proposal :

Remove from Annex B all interface standards. Gather them in a new Annex D. Complement each one with an explicative summary. The description of CCITT T.611 is contributed as an example.

The following is a tentative list being proposed:

- SC22:

POSIX

SQL

ISAM

DTR 10182: Guidelines for Language Bindings

CD 11404: Common Language - Independent Datatypes

SC22 N 1082: Common Language -Independent Procedure Calling Mechanisms

- SC24:

GKS

PHIGS

-CCITT:

T.611

Summary contributed for T.611:

T.611 API: a high-level API for Communication Software

T.611 is a standardized API situated at a high-level. It defines an interface between a user application requiring telecommunications services, called the Local Application, and a provider of these services, called the Communication Application. Examples of Local Applications (LAs) are word processors, spreadsheets and drawing applications. T.611 conforming Communication Applications (CAs) can provide services such as fax, teletex, telex, as well as other private services.

One or several CAs describe the services and their method of access in a special file called the Interface Configuration Environment or ICE. The format of this configuration file is defined in the T.611 standard, so is easily understood by any T.611-conforming LA.

The T.611 interface uses the client-server model: the LA passes requests to the CA, which then returns responses to the LA. The requests and responses are primitives called Task Data Descriptions (TDDs), and are in the form of character strings. The format of these character strings are defined by the API. The character format of the TDDs makes them machine-independent and allows them to be easily read using a simple text editor (useful for debugging and testing). TDDs exist for sending and receiving, for canceling and modifying send requests, for submitting printing and file conversion tasks, for dispatching received documents (in a multi- user environment), and for querying communication logs and queues.

The method of exchanging these TDDs is also described by the T.611 API. Several exchange method functions have been defined for initializing and terminating the LA-CA link, and for transferring TDDs between them. TDDs can be transferred using files or blocks of memory (the

- 11 -

- X/Open Preliminary Specification: EDI Messaging Package (XEDI)
- X/Open Preliminary Specification: Message Store API
- X/Open Preliminary Specification: ACSE/Presentation Services API
- X/Open Preliminary Specification: FTAM High-level API (XFTAM)
- X/Open Preliminary Specification: Management Protocols API (XMP)
- X/Open Guide: Distributed TP: Reference Model
- X/Open CAE Specification: Distributed TP: The XA Specification
- X/Open Preliminary Specification: Distributed TP: The TX (Transaction Demarcation) Specification
- X/Open Snapshot: Distributed TP: The Peer-to-Peer Specification
- X/Open Snapshot: The TxRPC Specification
- X/Open Snapshot: The XATMI Specification
- X/Open Snapshot: Data Management: SQL Call Level Interface (CLI)
- X/Open Preliminary Specification: SQL Remote Database Access