Business Plan for CEN XFS Workshop
(XFS- eXtensions for Financial Services)

1. Status of this Business Plan

Last adopted Business Plan: **XFS101290**

Incorporates action 40-1 (to update EURIGA in AURIGA)

2. Workshop Supporters

The proposers for the XFS Workshop were the now dissolved BSVC Consortium. The companies registered as Core Members in the XFS Workshop and having agreed on this Business Plan are the following:

- ATM JAPAN LTD
- AURIGA SPA
- CONFEDERACION ESPANOLA DE CAJA DE AHORROS
- DIEBOLD INCORPORATED
- DYNASTY TECHNOLOGY GROUP
- EASTERN COMMUNICATIONS CO. LTD
- FUJITSU ESPANA SERVICES
- GRG BANKING CO. LTD
- HESS CASH SYSTEM GMBH & CO. KG
- HITACHI OMRON TERMINAL SOLUTIONS CORP
- KAL
- KEBA AG
- NAUTILUS HYOSUNG INC.
- NCR FSG
- OKI ELECTRIC INDUSTRY SHENZHEN
- PHOENIX INTERACTIVE DESIGN INC.
- SALZBURGER BANKEN SOFTWARE
- SHENZHEN YIHUA COMPUTER LTD
- SIGMA SPA
- TALARIS PORTUGAL
- WINCOR NIXDORF INTERNATIONAL GMBH
- ZIJIN FULCRUM TECHNOLOGY CO.

The following companies were registered as Associate Members at the time of agreeing this Business Plan:

- BANK OF AMERICA
- ERNST REINER GMBH &CO. KG
- GLORY LTD
- LEVEL FOUR SOFTWARE LTD
- LEXCEL SOLUTIONS
- OKI ELECTRIC INDUSTRY CO LTD
- TECNOLOGIA BANCARIA SA

3. Workshop Objectives

The objective of WS/XFS is to provide an open and flexible framework for market players (manufacturers, service providers, users, etc.) to agree on next versions of the specifications under its authority, to cover new needs of the financial services marketplace. WS/XFS will consider requirements for additions to the specifications from any source and produce revised or additional specifications as appropriate.

The XFS committee is mindful of the PCI DSS regulations and PCI DSS regulations will be taken into account during the review of every XFS committee proposal. Compliance with PCI DSS however will be the responsibility of the individual implementation of the XFS specifications.

4. Workshop's Work Programme

4.1 Work in Progress

Work Item	Deliverable	Target Date
Further release Notes to Version 2.0	Versions of the release notes on version 2.0	Every meeting
Revisions to Version 3.0 of the XFS specification	Versions of the release notes on version 3.0	Every meeting
Revisions to Version 3.10 of the XFS specification	Versions of the release notes on version 3.10	Every meeting
Next point release of XFS 3.xx	CWA XXXXX (updating CWA15748) Point release with additional functionality	20XX

4.2 Work already delivered

CWA 13449 (XFS Version 2.0)

CWA 13849 (ActiveXFS Version 2.0)

CWA 14050 (XFS Version 3.03, XFS MIB 1.1)

CWA 15748 (XFS Version 3.10)

5. Workshop Organization

5.1 Membership

Membership (also called Registered Participants) in this Workshop is at company level. Companies can have as many representatives on the technical discussion lists or in meetings as they decide. All companies (Core members) have an equal weight in the decision process. Subsidiary companies, existing as a legal entity (with their own company name) can participate separately from the owning company in the technical work and in the decision process.

Registered Participants will exist in two categories:

- Core Participants
- Associated Participants

A description of both categories is below:

Core Participants will:

- Have an interest in infuencing the detail of the specification development or have an interest in the Workshop's activities as a major customer
- Decide on updates of the Business Plan
- Formally participate in the consensus building process and be recorded as such by the CEN Secretariat in relation to the resulting CWA(s)
- Contribute to the funding of the Technical Secretariat

Associated Participants will:

- Have an interest in the Workshop's activities as a customer feeding in requirements, rather than as a supplier of technical solutions
- Have equal access to XFS documents, with the exception of source code which will only be available to Core participants; In addition, WS-XFS meetings will be open to both Core and Associated Participants
- Only informally participate in the consensus building process i.e. they will not be recorded by the CEN Secretariat as supporting/objecting to the CWA(s) in question
- Contribute (but to a lesser extent than Core Participants do) to the funding of the Technical Secretariat

5.2 Addition of new members during the Workshop's lifetime

New members (core and associate) will be admitted as registered participants subject to their application being positively reviewed by the Core members. Criteria against which applications for new membership will be reviewed are:

- have products that (will) make use of the specifications under discussion or be a customer of the specifications
- subscribe to the objectives and targets in the business plan; if an application for membership leads to questions in this respect, then the business plan may be first written more explicit, clarifying its objectives and targets, before the new membership comes into effect

In addition, the new member has to agree with the funding arrangements for the Workshop Secretariat as agreed upon by the Workshop for the running year.

5.3 Support functions

5.3.1 Workshop Chairman and Vice-Chair

The function of Chairmanship is allocated for a duration of 2 years. The role of Chairman will be allocated to an individual, rather than to a company, in order to ensure neutrality. The initial 2 years period may be renewed as required. The chair shall be appointed by the core member companies by consensus. In case of non-consensus, appointment of the Chair is done by counting the number of Core member companies in support.

An election for the Chairman to be replaced can be called at any management meeting (see section 5.7).

The functions delivered by the Workshop Chairman shall be the following:

- Presiding Workshop plenary meetings.
- Ensuring that the Workshop delivers in line with the Business Plan.
- Management of the consensus building process
- Interfacing with the CEN and Workshop Secretariats regarding strategic directions, problems arising, external relationships, etc.
- Interfacing with the Workshop Secretariat on general Workshop management issues

Together with the Chairman's election, there will be the election of a Vice-Chair. The role of the Vice-Chair is to take over the responsibilities of the Chairman, when the Chairman is prevented from chairing.

5.3.2 Workshop Secretariat

The Workshop Secretariat is performed by a named individual representing a CEN National Member, in this case AFNOR.

The functions delivered by the Workshop Secretariat shall be the following:

- Organise and administer meetings
- Administer the Workshop’s email exploders, document repository and web pages

- Administer the requests routed to the WS/XFS – i.e. act as a help desk function to general WS/XFS issues and issues around submitting proposals to the WS/XFS, relaying them into the WS/XFS
- Produce meeting agendas, minutes and action/decision lists within 2 weeks of the meeting
- Act as an administrative contact point
- Chase actions on meeting members
- Ensure Business Plan updates are available as required

5.3.3 CEN Workshop Manager

CEN will task one of its staff as Workshop Manager to provide additional support to the XFS Workshop

The functions delivered by the Workshop Manager shall be the following:

- Set-up and maintain the electronic media used to publish the WS/XFS results
- Advise the Workshop in case of problems following interpretation of the Workshop rules
- Advise the Workshop on existing CEN procedures where necessary (e.g. publication of CWAs)
- Liaise with the Chairman and Secretariat in support of a fair interpretation of the consensus principles

5.4 Sub-groups

The XFS Workshop can establish other informal sub-groups to speed up the technical work. Sub-groups cannot decide, they only can prepare material for consideration/decision by the Workshop. Sub-groups are only visible inside the Workshop's internal working practices and have no formal existence. The Workshop Secretariat will maintain for each sub-group the following: subject area of the sub-group, the name of the sub-groups convenor and a sub-group email reflector. Furthermore, sub-groups have to organize their own secretariat support (reporting) during meetings. Subgroup meetings minutes should be available 4 weeks after meetings. In between meetings, they receive secretariat support similar to the XFS workshop itself.

5.5 Working Procedures

5.5.1 Governing principles

The governing principles are those for CEN Workshops in general. A number of additional working procedures specific to the XFS Workshop will apply, to guarantee a transparent process.

5.5.2 Consensus

Decisions are taken by consensus. It is important to understand that consensus does not mean unanimity. It is the responsibility of the Chairman (assisted by the Vice-Chair, Workshop Secretariat and if necessary the CEN secretariat) to judge whether consensus indeed has been reached. The chairman can decide to make use of indicative votes, as a means to assess the degree of consensus. In these indicative votes, the principle of "one vote for one core participant company" will apply. Proxy voting is accepted provided that this is announced to the mailing list in advance of the meeting. Consensus is considered to be reached if the level of support is bigger than or equal to 71% of the indicative votes cast.

5.5.3 Combining electronic participation with decisions at meetings

The need for consensus applies to decisions taken at meetings as well as to decisions resulting from discussions through email exchanges.

A necessary requirement is that the decisions - to the extent possible - also need to reflect the views of the registered participants that were not present at a meeting. This requires the timely announcement of the subjects that are for discussion and subsequent decision at meetings, as well as the timely availability of the supporting documentation (see below: the 4 week's notice principle).

Comments and proposals from registered participants unable to attend a meeting are then made available in advance to the electronic list and have to be considered by the meeting. The company having sent the written contribution but unable to attend, can ensure that its position at the meeting be represented by another XFS Workshop company, subject to an explicit notification to the Workshop's mailing list in advance of the meeting of this "proxy".

5.5.4 Management of Workshop Documents

The Workshop Secretariat maintains a list of current versions of the various parts of the CWA(s) under development. These current versions are known as "working documents".

Proposals for technical changes to these current working drafts shall be made via a specific "change proposal" document. When accepted the technical change will be incorporated in the next version of the above mentioned "working document". The Proposal and Working documents should not be in an encrypted format.

Proposals for new functionality have to be made in a specific proposal document, to be made available **4 weeks before** the meeting. When agreed by the Workshop, this proposal becomes a "working document".

Previously made proposals, for whatever reason refused or discussion postponed in earlier meetings, have to be explicitly resubmitted again as if they were new proposals, for the next meeting at which they are intended to be discussed.

5.5.5 Four week notice period

Meetings agendas shall be made available at least 4 weeks in advance. Agenda items for decision (with the material they reference) at a meeting have to be available **4 weeks in advance**

The minimum duration for a draft CWA to be available before endorsement by a Workshop is therefore also **4 weeks**. This does not apply to contributions/comments that relate to the CWA: they preferably are available before but also can be tabled at the meeting, depending on their size and level of technical detail.

Proposals for revising a Business Plan have to be available at least **4 weeks in advance**.

If the Workshop agrees, it can of course accept to discuss and decide in individual cases any contribution not meeting the minimum notice requirements.

5.5.6 Business Plan

The Business Plan contains the overview of current Workshop work items, with target dates for delivery. The Business Plan will be reviewed and, if necessary, updated and approved (by consensus) at every Workshop Management session (which is every 6 months). Proposals for revising a Business Plan have to be available at least 4 weeks in advance.

5.5.7 Minor Error Correction Procedure

In addition to the normal proposals for technical changes, there will also exist a procedure for minor changes such as typographical errors and formatting mistakes that may not deemed to be worthy of discussion. Examples of such minor changes could be:

- Case changes: ‘service provider’ should be ‘Service Provider’.
- Punctuation errors: missing or extra full stops, commas and quotes.
- Spelling mistakes.
- Incorrect formatting.

Examples of what does **not** constitute a minor change:

- Adding/removing an error code or parameter.
- Changing the functional meaning of a command, error code or event description.
- Subtle wording changes that affect behaviour, eg: “This is ~~not~~ persistent”.

The procedure for correcting minor changes will be as follows:

1. The proposal submitter posts to the subgroup forum explaining the proposed minor changes in order to invite comment. This does **not** need to have a formal proposal document attached, but should include a description of the changes and be simple enough to explain in one easy to understand paragraph. It should also include a cut and pasted paragraph from the working document showing clearly the corrections. The title of the forum posting should be prefixed with "MinorCorrection : " to distinguish it from normal proposal postings .

2. Three weeks from the date of posting will be allowed for review, any comments should be posted to the forum. If any objections are raised during this period then the proposal submitter will draft a formal proposal to the document register for discussion at the next scheduled subgroup meeting.
3. If no comments are received before the expiration of the three week deadline then the minor change is accepted.
4. After the three week deadline the proposal submitter is then responsible for updating and submitting a new working document with the change incorporated, **unless** the working document is already scheduled to have other agreed changes incorporated into it - in this case the proposal submitter should contact the person responsible for updating the document and ask them to include the change for the next document revision. Again this should be done using the forum.
5. The working document is updated and submitted to the document register for formal adoption at the next technical meeting.

5.6 XFS Workshop Management Session

Every other meeting (thus typically every 6 months), the XFS Workshop meeting will include a Management Session, to which the following responsibilities are delegated:

- decision on new technical directions to be taken by the Workshop (which should then also be recorded in the Workshop's business plan)
- approval of Workshop business plan (work items, target dates for delivery, internal procedures, etc)
- decision on way forward in case of blockages on technical issues
- issues related to compliance with the CWAs
- other

Decisions in the remit of the Management Sessions can also be taken electronically.

In exceptional cases, the Chairman has the possibility to convene a management session to take place at the next Workshop meeting (thus to fit in the quarterly meeting pattern of the Workshop, rather than on a 6-monthly basis).

5.7 Availability of Deliverables

The CWAs delivered by the XFS Workshop will be freely downloadable from the CEN web-site. XFS Core members will pay a contribution to CEN to compensate CEN National Standard Bodies who cannot make income from the sale of the CWAs.

5.8 Meeting Reports

A report of the Workshop-meetings and of its working groups, highlighting the decisions taken at the meeting, have to be circulated to the Workshop registered participants within 2 weeks.

6. Resources

The cost of the operation of the XFS Workshop Secretariat (by AFNOR, with some overhead for CEN) will be entirely borne by contributions from the Workshop registered participants (core and associate).

The Workshop's participants will provide the necessary voluntary resources for functions as editor, chair, sub-group leader, etc.

For 2010, the contributions are therefore agreed as follows:

- 3480 Euros for Core members (of which 1000 Euro is to continue the free availability of the CWAs on CEN web-site)
- 1225 Euros for Associate members

7. Related activities and Liaisons

None

8. Contact Points

Clément Chevauché, XFS Workshop Secretary, AFNOR

clement.chevauche@afnor.org

Stuart Currie, XFS Workshop Chairman,

Stuart.Currie@diebold.com

John Byers, XFS Workshop Vice-Chairman,

JB514933@ncr.com

9. Document Management Proposal approved during the meeting held on September 25, 2002.

When a new proposal for the next version is approved by a subgroup, the master device specification is updated, and put forward for approval at the next main technical meeting. However, each proposal includes a 'Proposal Background' & Background Compatibility section that is useful in tracking the reason for the change, and the impact on application developers. This information should not be lost when updating the master document.

This proposal requests that, for each new piece of functionality added to each device spec, these sections are also added to the master document. This way we can track all the functional changes in the 'master' device documents as they are added.

The master document would only be updated when a proposal has been agreed by the device subgroup & then forwarded to the main group for approval

Before we release the next version, we can either move this info to the migration documents, or keep in the main spec.

For example a proposal (*Simplified Application dialog with Chip Cards*) has been approved by the IDC subgroup and has been forwarded to the main technical group for formal adoption for the next release.

The document Introduction section has been adapted to have the following structure

1. Introduction
    1.1 Simplified Application dialog with Chip Cards
        1.1.1 Proposal Background
        1.1.2 Backwards Compatibility
    1.2 <Next Agreed Proposal>
        1.2.1 Proposal Background
        1.2.2 Backwards compatibility
    . . .

11. CEN XFS Release Framework and Backwards Compatibility Definitions

CEN XFS Release Framework

The following table defines the types of CEN XFS release possible, when they are appropriate, the numbering scheme for each release type and the backwards compatibility requirements for each release type.

Release type	Description	Numbering Convention	Backwards Compatibility Rules
Fast-Track	1. A fast track release is driven by regulatory requirements. 2. Only a sub-set of device class specifications are affected (typically between 1 and 5). 3. Unaffected device classes remain unchanged in content and version number. 4. Fast-track releases do not contain proposals that do not conform to the relevant Backwards Compatibility Rules. 5. Fast Track releases do not contain new device classes	1. The overall current XFS version number is incremented by a one hundredth, e.g. XFS 3.03 would be incremented to 3.04, 2. The modified device classes are incremented to match the new overall XFS version number. E.g if the SIU was modified in a theoretical 3.04 fast-track its version would jump from 3.01 to 3.04. 3. Unmodified device class versions numbers do not change.	Fast-Tracks releases are controlled by the Point-Release Backwards Compatibility Rules.

Point Release	1. A point release is a scheduled CEN XFS release. It contains all proposals that are available for release at the scheduled release cut-off. The frequency of his type of release is determined by the CEN XFS Management committee taking customer expectations, proposal availability, proposal importance and other relevant factors into account. 2. All device classes will be modified. Even device classes with no functional changes will have their version number increased. 3. Point releases do not contain proposals that do not conform to the relevant Backwards Compatibility Rules. 4. A point-release can contain a new device class. 5. Point Release should be released as fast track numbers are exhausted.	1. The overall current XFS version number is incremented to the next tenth, e.g. XFS 3.03 would be incremented to 3.10. 2. All device classes are incremented to match the new overall XFS version number, even if there is no functional change	Point-Releases are controlled by the Point-Release Backwards Compatibility Rules.

Major Marketing	An upgrade of the major number of standard version for non technical reasons, for example: 1. To-coincide with some other industry event, e.g release of a new operating system 2. Periodic upgrade to baseline the XFS functionality 3. Run out of point release numbers, e.g 3.90 has been reached	1. The Major release number would be increased by one and the minor number would be re-set to zero, e.g 3.12 would become 4.00. 2. All device classes are incremented to match the new overall XFS version number, even if there is no functional change	Major Marketing releases are controlled by the Point-Release Backwards Compatibility Rules
Major Functional	A major release is less clearly defined but would be influenced by the following: 1. Removal of a device class from XFS. 2. Addition of a new fundamentally crucial device class. 3. Change in underlying operating system technology that affects the interface or configuration mechanisms. 4. Change in the CEN XFS interface technology. 5. Break in Backwards Compatibility	1. The Major release number would be increased by one and the minor number would be re-set to zero, e.g 3.12 would become 4.00. 2. All device classes that remain within the CWA will be upgraded to the new version number.	Major-Releases are controlled by the Major-Release Backwards Compatibility Rules.

Backwards Compatibility Definition

The method for controlling backwards compatibility within the CEN XFS standard development process is based in the principles of Version Negotiation and Rules Based Proposal Evaluation.

Whenever a session is opened with a service provider, applications and service providers must negotiate to a common supported version. Once this version is agreed both parties must communicate using structures and messages that are compatible with that version. To access fast-track functionality applications may have to negotiate different versions with each device class.

During the XFS standard development and enhancement process, all proposals must be evaluated against the Rules defined for the type of release in which it is to be included. If a proposal does not meet the stipulated rules it will not be included in the release.

These two principles maintain the XFS architecture integrity and

- Provides a framework for proposal to be developed against.
- Makes migration from one version of XFS to another simpler
- Defines a set of rules against which proposals can be checked
- Does not proliferate multiple commands with the same functionality
- Provides a simple to use, high performance API

Version Negotiation

CEN XFS has had the principle of version negotiating at its core from the very first version of the standard. In summary, an application specifies a minimum and maximum version range that it can support, and the service provider does the same. Then, when a session is opened a negotiation process takes place to agree on common version that both can support. Once the version has been negotiated, the application and service provider must conform to the agreed interface. This means that both must be able to handle messages conforming to the agreed version and that both must only generate output that conforms to the agreed version. A service provider must not generate output that is incompatible with the negotiated version.

Rules Based Proposal Evaluation

Although versioning provides a stable version that both parties conform to at any point in time, it does not on its own help an application migrate forward from one version of the standard to the other. This can be seen in the problems applications (and service providers) have encountered when migrating from CEN XFS 2.x to

CEN XFS 3. When the XFS 3 specification was defined a number of changes were made that made migration unnecessarily difficult.

All proposals must now adhere to the rules defined in the following table before they will be accepted in to the release under consideration.

In order to generalise the rules the XFS Service interfaces has been broken into the following components.

- Input Structures
- Input field values / literals
- Output structures
- Output field values / literals
- Error Codes
- Events

	Point Release	Point Notes	Major Release	Major Notes
Extend Input Structures	No	Input structures cannot be extended as this would force an application change.	Yes	
Extend Command Completion Output Structures	Yes – but only at the very end of the structure and under the following conditions: 1) Existing apps will still function correctly without having to read the new fields. 2)The structure is not directly embedded within another structure, ie it is referenced by a pointer rather than being directly referenced	Output structures can be extended at the end as applications that do not require the new functionality can use the previous structure without modification.	Yes, but only at the very end of the structure.	

Extend Event Output Structure	Yes – but only at the very end of the structure and under the following conditions: 1) Existing apps will still function correctly without having to read the new fields. 2)The structure is not directly embedded within another structure, ie it is referenced by a pointer rather than being directly referenced	As for Command Completion event above.	Yes, but only at the very end of the structure.	
Re-name literals or change their values	No		No	
Remove Fields from Input, Output or Event Structures	No	This would modify the offset of the other fields in the output structure requiring the application to provide version logic whenever a field is accessed.	No	
Add new Input Literals	Yes	Existing apps will not and need not use the new values to maintain existing behaviour.	Yes	

Add new Output Literal Values	Yes	The API specification should recommend that a defensive programming style is adopted when processing output flags / fields with defined values as they will be extended in the future.	Yes	The API specification should recommend that a defensive programming style is adopted when processing output flags / fields with defined values as they will be extended in the future.
Generic Error Codes	Yes	Adding generic errors should be avoided where possible as it can effect every command in every device. Again a defensive programming style should be recommended.	Yes	
Add new Command Specific Error Codes to existing commands	Yes	Recommend a defensive programming style, i.e. do not crash if unexpected error code received.	Yes	Recommend a defensive programming style, i.e. do not crash if unexpected error code received.
Add new Command Specific Error Codes to new commands	Yes	No impact on existing applications.	Yes	No impact on existing applications.

Add new Events to existing commands	Yes, but only if existing applications can still function without the additional functionality represented by the new event	Need to be careful that existing applications will not need to process the event to get the original behaviour. If an application would have to process the event to get the original behaviour a new command should be added.	Yes	
Add new Events to new commands	Yes	No impact on existing applications.	Yes	No impact on existing applications.
New Commands	Yes	No impact on existing applications.	Yes	No impact on existing applications.
Add Input Parameter to command that previously had no input, i.e. change Input Parameter from Null to lpStruct	Yes	Must also allow NULL to be passed to get the original functionality.	Yes	Must also allow NULL to be passed to get the original functionality.
Add Output Parameter to command that previously had no output, i.e. change output from Null to lpStruct	Yes, but only if existing applications are un-affected.	Original command behaviour must still be possible.	Yes	

Maintain Standardised lpszExtra	Yes, but maintain the previous standardised lpszExtra fields and also duplicate the information within the main structures according to rules above.	.	No	.
Extend Standardised lpszExtra	No	This field is for vendor specific functionality.	No	This field is for vendor specific functionality.