

X/Open Snapshot

1

2

Manageability Guidelines:

3

Special Draft Sept 1995 for XTP TWG review

4

X/Open Company Ltd.



5

6

© *September 1994, X/Open Company Limited*

7

All rights reserved. No part of this publication may be reproduced, stored in a retrieval system,
8 or transmitted, in any form or by any means, electronic, mechanical, photocopying, recording or
9 otherwise, without the prior permission of the copyright owners.

10

X/Open Snapshot

11

Manageability Guidelines: Special Draft Sept 1995 for XTP TWG review

12

X/Open Document Number: S426

13

Published by X/Open Company Ltd., U.K.

14

Any comments relating to the material contained in this document may be submitted to X/Open
15 at:

16

X/Open Company Limited

17

Apex Plaza

18

Forbury Road

19

Reading

20

Berkshire, RG1 1AX

21

United Kingdom

22

or by Electronic Mail to:

23

XoSpecs@xopen.org

Contents

25	Chapter 1	Introduction.....	1
26	1.1	Requirements.....	1
27	1.2	Scope.....	2
28	1.3	Terminology.....	2
29	1.4	Applications.....	2
30	1.5	Background.....	2
31	Chapter 2	Management Requirements	3
32	2.1	Management Tasks	3
33	2.2	Resources.....	3
34	2.3	Services	3
35	2.4	Relationship to Other Specifications	4
36	Chapter 3	Overview of Management Issues	5
37	3.1	Management Goals.....	5
38	3.2	Management Requirements.....	6
39	3.3	Standards.....	6
40	3.3.1	Existing Standards.....	6
41	3.3.2	Emerging Standards.....	6
42	Chapter 4	Management Specification	7
43	4.1	Management Tasks	7
44	4.2	Resources.....	7
45	4.3	Services	8
46	Appendix A	Template	11
47	Chapter n	Management Requirements	12
48	n.1	Overview of Management Requirements	12
49	n.1.1	Management Goals.....	12
50	n.1.2	Management Requirements.....	12
51	n.1.3	Standards.....	12
52	n.1.4	Emerging Standards.....	12
53	n.2	Management Requirements.....	12
54	n.2.1	Management Tasks	12
55	n.2.2	Resources.....	12
56	n.2.3	Services	12
57	n.2.4	Relationship to Other Specifications	13
58		Index.....	15

X/Open

X/Open is an independent, worldwide, open systems organisation supported by most of the world's largest information systems suppliers, user organisations and software companies. Its mission is to bring to users greater value from computing, through the practical implementation of open systems.

X/Open's strategy for achieving this goal is to combine existing and emerging standards into a comprehensive, integrated, high-value and usable open system environment, called the Common Applications Environment (CAE). This environment covers the standards, above the hardware level, that are needed to support open systems. It provides for portability and interoperability of applications, and so protects investment in existing software while enabling additions and enhancements. It also allows users to move between systems with a minimum of retraining.

X/Open defines this CAE in a set of specifications which include an evolving portfolio of application programming interfaces (APIs) which significantly enhance portability of application programs at the source code level, along with definitions of and references to protocols and protocol profiles which significantly enhance the interoperability of applications and systems.

The X/Open CAE is implemented in real products and recognised by a distinctive trade mark — the X/Open brand — that is licensed by X/Open and may be used on products which have demonstrated their conformance.

X/Open Technical Publications

X/Open publishes a wide range of technical literature, the main part of which is focussed on specification development, but which also includes Guides, Snapshots, Technical Studies, Branding/Testing documents, industry surveys, and business titles.

There are two types of X/Open specification:

- *CAE Specifications*

CAE (Common Applications Environment) specifications are the stable specifications that form the basis for X/Open-branded products. These specifications are intended to be used widely within the industry for product development and procurement purposes.

Anyone developing products that implement an X/Open CAE specification can enjoy the benefits of a single, widely supported standard. In addition, they can demonstrate compliance with the majority of X/Open CAE specifications once these specifications are referenced in an X/Open component or profile definition and included in the X/Open branding programme.

CAE specifications are published as soon as they are developed, not published to coincide with the launch of a particular X/Open brand. By making its specifications available in this way, X/Open makes it possible for conformant products to be developed as soon as is practicable, so enhancing the value of the X/Open brand as a procurement aid to users.

• *Preliminary Specifications*

These specifications, which often address an emerging area of technology and consequently are not yet supported by multiple sources of stable conformant implementations, are released in a controlled manner for the purpose of validation through implementation of products. A Preliminary specification is not a draft specification. In fact, it is as stable as X/Open can make it, and on publication has gone through the same rigorous X/Open development and review procedures as a CAE specification.

Preliminary specifications are analogous to the *trial-use* standards issued by formal standards organisations, and product development teams are encouraged to develop products on the basis of them. However, because of the nature of the technology that a Preliminary specification is addressing, it may be untried in multiple independent implementations, and may therefore change before being published as a CAE specification. There is always the intent to progress to a corresponding CAE specification, but the ability to do so depends on consensus among X/Open members. In all cases, any resulting CAE specification is made as upwards-compatible as possible. However, complete upwards-compatibility from the Preliminary to the CAE specification cannot be guaranteed.

In addition, X/Open publishes:

• *Guides*

These provide information that X/Open believes is useful in the evaluation, procurement, development or management of open systems, particularly those that are X/Open-compliant. X/Open Guides are advisory, not normative, and should not be referenced for purposes of specifying or claiming X/Open conformance.

• *Technical Studies*

X/Open Technical Studies present results of analyses performed by X/Open on subjects of interest in areas relevant to X/Open's Technical Programme. They are intended to communicate the findings to the outside world and, where appropriate, stimulate discussion and actions by other bodies and the industry in general.

• *Snapshots*

These provide a mechanism for X/Open to disseminate information on its current direction and thinking, in advance of possible development of a Specification, Guide or Technical Study. The intention is to stimulate industry debate and prototyping, and solicit feedback. A Snapshot represents the interim results of an X/Open technical activity. Although at the time of its publication, there may be an intention to progress the activity towards publication of a Specification, Guide or Technical Study, X/Open is a consensus organisation, and makes no commitment regarding future development and further publication. Similarly, a Snapshot does not represent any commitment by X/Open members to develop any specific products.

Versions and Issues of Specifications

As with all *live* documents, CAE Specifications require revision, in this case as the subject technology develops and to align with emerging associated international standards. X/Open makes a distinction between revised specifications which are fully backward compatible and those which are not:

- a new *Version* indicates that this publication includes all the same (unchanged) definitive information from the previous publication of that title, but also includes extensions or additional information. As such, it *replaces* the previous publication.

- a new *Issue* does include changes to the definitive information contained in the previous publication of that title (and may also include extensions or additional information). As such, X/Open maintains *both* the previous and new issue as current publications.

Corrigenda

Most X/Open publications deal with technology at the leading edge of open systems development. Feedback from implementation experience gained from using these publications occasionally uncovers errors or inconsistencies. Significant errors or recommended solutions to reported problems are communicated by means of Corrigenda.

The reader of this document is advised to check periodically if any Corrigenda apply to this publication. This may be done either by email to the X/Open info-server or by checking the Corrigenda list in the latest X/Open Publications Price List.

To request Corrigenda information by email, send a message to info-server@xopen.co.uk with the following in the Subject line:

request corrigenda; topic index

This will return the index of publications for which Corrigenda exist.

This Document

This document is a Snapshot (see above). It is one of several documents within X/Open's Systems Management programme (XSM).

This document provides guidelines for writing a chapter on Management Requirements for inclusion in an API specification.

It is intended that this Snapshot will be updated based on feedback from users of this document.

The guidelines are structured as follows:

- Chapter 1 describes how the guidelines should be used.
- Chapter 2 introduces the concepts used in the rest of the document
- Chapter 3 outlines the contents of an overview of the management requirements
- Chapter 4 provides more detailed guidance on the specification of management requirements.
- Appendix A contains a template for a management requirements chapter.

Referenced Documents

171

172

The following document is referenced in this guide:

173

XIMS

174

X/Open Snapshot, May 1992, Systems Management: Identification of Management Services

175

XRM

176

X/Open Guide, August 1993, Systems Management: Reference Model

Introduction

This document discusses issues concerned with introducing "manageability" into the definition of application programming interfaces (APIs). This document provides guidance from the X/Open Systems Management Working Group as to how resources that are made available via a new API may be managed, with the goal that this management should be largely compatible with that applied to other resources.

The primary audience for this document is intended to be X/Open Technical Working Groups active in areas other than Systems Management, e.g. Transaction processing, Security, etc. However, the basic principles described in this document are equally applicable to any interface specification activity, and this document is therefore being made available to a wider audience as an X/Open Snapshot.

1.1 Requirements

While it is clearly possible to specify an API without addressing issues of management, as can be seen from many existing interfaces, it is generally much simpler to introduce management at the beginning of the process than to retro-fit it afterwards.

The requirement to be "manageable" has considerable influence on what information is made available, and the way in which different information elements are grouped together. When management issues are addressed in the context of an existing interface, it may be necessary to create new information elements to contain management information. In addition, the management information may be scattered among many existing interfaces in a way that makes a very untidy mapping between the interface and the tasks that the Administrator needs to perform.

This document imposes no absolute requirement to include a management chapter within all X/Open specifications, nor does it mandate the precise form that such a chapter, if present, should take. It does however strongly encourage the addressing of management issues from the outset of the specification development process. It does introduce a set of concepts and a corresponding vocabulary, which can be used to describe management issues in a common way. It also provides a suggested template that could be used to provide a management chapter.

The decision not to adopt a prescriptive approach reflects a belief that in generally the specialised knowledge necessary to describe the management aspects of a particular interface resides within the group undertaking the work, and it is more important that the group should be able to describe those aspects in a way that comes naturally to them. The provision of a relatively simple set of concepts is intended to structure the description in such a way that it can be subsequently mapped into a common management framework.

1.2 Scope

The purpose of a management section within an interface specification is to outline the management requirements of that interface. It should describe the tasks that an Administrator needs to be able to perform in order to manage the resources to which the interface provides access, and the management information that needs to be stored in order to implement those tasks.

In many cases, at least some of the management tasks are likely to be concerned with controlling access to the resources. As a result of this, the management issues are likely to be closely connected to the interface's security issues.

1.3 Terminology

The terminology in this document is taken from the X/Open Systems Management Reference Model (see reference **XRM**).

1.4 Applications

There are two types of specifications to which this document might be applied:

- Existing specifications which do not include any material addressing management issues. In this case, management concerns can either be introduced into a convenient update of the specification, or a new, separate document can be produced. The choice between these two options is likely to be influenced by the update schedule for the specification, and the anticipated size of the management aspects of the specification.
- Specifications under development, in which case, management should be addressed in the document at the time of its development.

1.5 Background

Management is an essential part of information systems technology, and this is particularly true within the world of open systems.

Within a world of heterogeneous, distributed systems, provided by many different vendors, it is of paramount importance that they implement a common model of interoperable management.

Within the context of X/Open specification development, management is primarily concerned with the provision of common interfaces and common information. The principal focus is on programmatic and command line interfaces, enabling the development of graphical interfaces as a point of differentiation between vendors.

Management Requirements

This chapter presents the basic concepts that are used within the X/Open Systems Management programme. They are drawn from the Systems Management Reference Model (see reference XRM).

At a fundamental level, there are Administrators, performing "Management Tasks", who exercise some form of control and/or wish to be kept informed about the operation of one or more "Resources".

The Reference Model does not specify what Management Tasks or Resources exist. It describes the framework within which they can be realised, a set of "Services" that provide the functionality necessary to allow Management Tasks and Resources to communicate.

2.1 Management Tasks

Management Tasks represent the management activities performed by Administrators. As such, they are abstract entities which make use of the underlying functionality of the management system in order to achieve the desired action.

Management Tasks do not appear explicitly within the Reference Model. However, they are the essential functionality that the Model exists to support. The Model provides the framework necessary to support the various components that need to exist in order to implement the operations that are represented by Management Tasks.

An example of a Management Task is adding a user, which involves the creation and manipulation of several different Resources within the system.

2.2 Resources

Resources are the entities within a system or network of systems that require management. Resources can include physical entities (such as printers or routers) as well as logical entities (such as users or groups). Not all resources require management.

2.3 Services

Services exist to provide the common facilities that must be provided in order to support distributed systems applications. Services can be divided into three major classes.

- General Services,
- Management Services, and
- Application Services.

This classification derives from the relationship of a specific service to the specific problem space being addressed.

General Services are characterised as being of use to a wide range of different problem areas.

Management Services are common facilities which have been specialised for distributed management. Areas of specialisation might include: policies for more centralised control of

99 security, policies for configuring and distributing applications, and the ability to control the
100 location of objects.

101 Application Services are services that are specific to some particular functional area within the
102 overall management problem space. While these services are not of general use to a wide range
103 of management applications, they provide common services to implementations addressing a
104 particular area. An example might be a catalogue service provided for the use of multiple
105 backup and restore applications.

106 A fuller discussion of management and general services is given in the X/Open Systems
107 Management Identification of Management Services Snapshot (see reference XIMS).

108 **2.4 Relationship to Other Specifications**

109 Specifications should clearly identify management dependencies with other specifications in the
110 following ways:

- 111 • The specification may depend on the management capabilities contained within some other
112 specification. This is likely to be because it is highly dependent on resources that are
113 described in the other specification.
- 114 • The specification may be a provider of resources, and their management, to other
115 specifications.

Overview of Management Issues

This chapter discusses the information that should be provided in the introductory analysis of the management requirements of a specification. It describes section 1 of the model management requirements chapter contained in the appendix.

3.1 Management Goals

The specification should clearly identify the goals that it seeks to achieve in identifying the management requirements of that specification. These may include:

- Access Control - There may be a requirement to control access to the resources to which the API provides access. Such a requirement is clearly closely related to the security requirements of the specification, which should define precisely how access control should be managed.
- Configuration Management - It may be necessary to provide configuration information that is necessary to the correct functioning of the API. This information may not be directly accessed by an application using the API, but will condition the way in which it operates.
- Performance Management - There may be a requirement to both measure and manage the performance aspects of the specifications. This may include issues such as response times, usage of resources such as storage and processing cycles, I/O rates, etc.
- Fault Management - When faults occur, it may be desirable to report on the cause of the error conditions, generate event reports to bring the fault to the notice of an administrator, or to isolate a resource so that it is accessible only to management applications until the error has been corrected.
- Accounting Management - It may be necessary to generate usage information about the use of the API and its underlying usages so that appropriate charges can be made for their use.

An important goal should be the ability to define and manage policy. Different organisations are likely to have different requirements for the operation of the functionality provided by a specification. For instance, there may be different requirements for default access controls, or for the default configuration of new instance of a resource. In addition to being able to specify to define such policies, it must also be possible to override such policies when necessary, subject to appropriate authorisation.

3.2 Management Requirements

The specification should provide an overview of the detailed requirements.

It is also important to be aware of the different classes of users who require access to management interfaces. These include:

- Computer System Administrators (User)
- Management Application Developers (Developer)
- System Service Providers (Service Provider)
- Computer System Resource Planners (Enterprise)

The requirements of these different constituencies differ both in the degree of detail that needs to be exposed and in the types of operations that need to be performed. These differing requirements also affect the types of interface that need to be provided. Administrators need Command Line Interfaces (CLIs) whereas Developers require Application Programming Interfaces (APIs).

3.3 Standards

3.3.1 Existing Standards

Identify any management standards that may be used to support the implementation of the management requirements of this specification.

3.3.2 Emerging Standards

Identify any applicable management efforts currently underway that may be relevant to this specification now or in the future.

Management Specification

This chapter provides illustration as to how the concepts introduced in this document might be applied. It describes section 2 of the model management requirements chapter contained in the appendix.

4.1 Management Tasks

When identifying the Management Tasks appropriate to a particular specification, it is necessary to take on the viewpoint of a system administrator. The key goal is to focus on the tasks that an administrator actually needs to perform in order to manage the resources.

In general, there is a one-to-many relationship between Management Tasks and the resources that are being managed. One Management Task will perform operations on many individual resources in order to accomplish the action that the administrator wishes to perform.

If, instead of taking the administrator's viewpoint, a resource-oriented viewpoint is adopted, it is much more likely that a one-to-one relationship between tasks and resource operations will develop. This leads to an explosion of Management Tasks which exhibit too fine a degree of functionality, requiring the administrator to perform sets of tasks in order to achieve the desired action.

A good example is to consider the topic of user account management. In order to add a user it is necessary to perform operations on a wide variety of resources (files, authorisation databases, etc). The order in which some of the operations are performed is important, while others can be performed at any point in the sequence. This sequencing requirement may vary between different implementations, severely impacting the portability of Administrators between systems. Trying to remember precisely which operations must be performed in what order on each system is a complex task, fraught with risk of failure, and actively interferes with the possibility of successful management.

The task that the Administrator needs to perform is simply "to add a user", regardless of the precise sequence of operations that need to be performed in any particular environment.

4.2 Resources

All specifications provide access to resources within an information system. These resources may be actual physical entities, such as disks or processors, logical entities, such as files and processes, or abstract entities such as users and groups. APIs allow application developers to make use of the capabilities of these resources to store and process information, to control access to information, and to make information available in external forms via networks, printers, and other media.

The resources contained within a specification may vary in complexity. For example, files are a relatively simple entity, supporting a simple set of functions, open/close/read/write. However, the data contained within a file may be a very complex entity, requiring a much richer set of functions, such as indexed access methods and data versioning.

It is important to be clear precisely which resources a specification is concerned with, and to separate them from other resources which are defined elsewhere. Thus a security specification might be concerned with managing Access Control Lists, providing an interface to manipulate

ACLs, without providing direct access to the underlying storage mechanisms, (probably files), which are used to store the actual data.

In such a case, the management concerns of the specification are centred on the data that the files contains, rather than the files that contain the data. There is clearly a dependency on some other mechanisms to be able to manipulate the actual storage, but this is not reflected directly in the specification.

In many cases there may not be specific management interfaces provided in order to manipulate a resource. Management operations are performed using the normal functional interface that is used by application writers who are using the API. The difference lies in the purpose for which the operation is performed - the "role" being taken by the software performing the operation. In order to perform a particular Management Task, it may not be necessary to provide any additional interfaces beyond those that are already contained within the specification. However, in some cases there may be a need to provide additional management interfaces in order to perform functions that are solely used in order to manage the resource. An example of this is an interface that enables a printer queue to accept print jobs.

4.3 Services

Application developers should consider addressing manageability requirements when designing their application. Instrumenting their application for manageability may be best performed via service extensions to X/Open technology specifications, eg. offering a backup capability in a transaction processing API. The following services may also be included in an appendix of an X/Open technology specification as guidance to application developers.

Management Services for Consideration:

Installation An application is likely to need to be modified to support a software distribution mechanism. It is suggested the POSIX 1387.2 be used for this purpose. Also, it should be noted X/Open SysMan is in the process of developing an interoperability standard for POSIX 1387.2 software distribution.

Licensing If software licensing is desirable for revenue protection, the application will likely need extensions to incorporate the mechanism. A standard licensing mechanism will insure consistency. This is likely to be an X/Open SysMan work item in the near future.

Backup-Restore An application which is collecting or generating critical data may want to backup the data for safety. X/Open SysMan is currently developing APIs for data backup and restore for such an application.

Logging Within existing systems, the *syslog* function is widely used by a variety of functional areas in order to satisfy their requirements for a logging mechanism. This has the added benefit of gathering information into a central pool, easing the task of determining precisely what occurred when something goes wrong.

Performance An application or system with time critical requirements can use the performance interfaces and data definitions to provide critical information to a systems administrator. X/Open SysMan is in process of standardizing the Universal Measurement Architecture which contains interfaces for collection and management, as well as specific resource definitions.

249	Configuration	Configuration of applications can require significant time just to get the
250		application usable. Standardized and consistent configuration can help
251		minimize this overhead. A group or class of applications should consider
252		using a common method of configuration.
253	Event Service	An application or system may generate conditions and events that need
254		to be monitored and logged, and necessary notification given to systems
255		administrators. An event service is being investigated by the X/Open
256		SysMan working group. Enabling automation of an action associated
257		with events is the key aspect to move to a proactive mode of operation.
258	Accounting	Accounting services can provide applications and systems the ability to
259		bill for usage and collect historical data for capacity planning. There are
260		ISO standards which address this requirement, and likely other standards
261		groups will address this in the future.
262	When considering the management requirements of a specification, common services should be	
263	identified wherever possible. If there is no existing service specification, the requirement should	
264	be clearly identified.	

Template

265

266

This is a template for a chapter covering management requirements in X/Open specifications.

267

Explanatory text is marked with shading in the margin.

268

For additional information on how to produce a management chapter, see the remainder of this document.

269

Management Requirements

270

271 This chapter defines the management requirements of this specification. The issues are
 272 discussed in terms of Management Tasks, Resources, and Services, and the impact on other
 273 specifications. An overview and details of the management requirements are provided.

274 n.1 Overview of Management Requirements

275 n.1.1 Management Goals

6 Identify management goals.

277 n.1.2 Management Requirements

8 Provide an overview of the management requirements.

279 n.1.3 Standards

0 Identify applicable management standards that may be relevant to this specification.

281 n.1.4 Emerging Standards

2 Identify applicable management efforts currently underway that may be relevant to this
 3 specification now or in the future.

284 n.2 Management Requirements

285 n.2.1 Management Tasks

6 Identify the Management Tasks due to the introduction of new resources or capability from this
 7 specification.

288 n.2.2 Resources

9 Identify the underlying resources contained within this specification that need to be managed.

290 n.2.3 Services

1 Identify the management services required or provided by this specification.

292 **n.2.4 Relationship to Other Specifications**

3 Identify the management dependencies between this specification and other specifications.

1 ***/ Index***

2

