

中华人民共和国民用航空行业标准

MH/T 0048.1—2014

民用机场共用旅客处理系统技术规范 第1部分：系统结构

Specification of common use passenger processing systems in civil aviation airports
Part 1: System architecture

2014 – 12 – 03 发布

2015 – 03 – 01 实施

中国民用航空局 发布

目 次

前言 II

1 范围 1

2 规范性引用文件 1

3 术语和定义 1

4 CUPPS 总则..... 2

5 CUPPS 系统功能..... 3

6 平台管理 14

7 工作站管理 19

8 用户管理 26

9 应用程序管理 32

10 设备管理 37

11 应用程序需求 43

前 言

MH/T 0048分为三个部分：

- 第 1 部分：系统结构；
- 第 2 部分：应用软件数据交换；
- 第 3 部分：硬件设备数据交换。

本部分为MH/T 0048的第1部分。

本部分按照GB/T 1.1-2009给出的规则起草。

本部分由中国民用航空局人教司提出。

本部分由中国民用航空局航空器适航审定司批准立项。

本部分由中国民航科学技术研究院归口。

本部分起草单位：中国民航大学、中国民航信息网络股份有限公司。

本部分主要起草人：李建伏、张博、徐涛、赵森、贺怀清、董晨欣、张清文、惠康华、丁玎。

MH

民用机场共用旅客处理系统技术规范

第1部分：系统结构

1 范围

MH/T 0048的本部分规定了共用旅客处理系统的平台管理、工作站管理、用户管理、应用程序管理、设备管理以及对应用程序的要求。

本部分适用于中国民用机场共用旅客处理系统的建设。

2 规范性引用文件

下列文件对于本文件的应用是必不可少的。凡是注日期的引用文件，仅所注日期的版本适用于本文件。凡是不注日期的引用文件，其最新版本（包括所有的修改单）适用于本文件。

IATA推荐性惯例1797：2011 共用旅客处理系统技术规范 (Recommended practice 1797 common use passenger processing systems technical specification)

标准编号欧洲航空协会技术规范2009 (AEA Technical Specifications 2009)

3 术语和定义

下列术语和定义适用于本文件。

3.1

共用旅客处理规范 CUPPS Common Use Passenger Processing Systems

由国际航协定义，用于航空公司使用机场的终端设备的信息处理规范。

3.2

CUPPS工作站 CUPPS Workstation

运行于CUPPS平台上的硬件和操作系统软件。硬件包括计算机、移动终端设备、智能手机、简易客户终端等。

3.3

CUPPS应用 CUPPS Application

运行在CUPPS平台上，使用CUPPS平台接口的应用程序。

3.4

商用现成品或技术 COTS Commercial off the shelf

指可以采购到的具有开放式标准定义的接口的软件或硬件产品。

COTS应用即基于COTS的应用。

4 CUPPS 总则

4.1 总体要求

CUPPS平台应遵循以下原则：

- 平台独立性：不依赖于平台提供商，也不必对不同提供商提供的不同平台进行分别维护；
- 简化业务流程：提供一个灵活的平台，能简化多种业务的应用；
- 功能定义范围：本部分中定义的为必须实现的功能；
- 经济效益：通过简化产品的安装，支持与升级维护，降低航空公司与机场的重复投资；
- 可维护性：提供强有力的可维护性，平台需要可以快速的部署、维护和升级；
- 安全性：提供相应的安全措施确保系统可以平稳运行；
- 服务性：可以方便地应对应用程序的升级和维护需求。

适用于CUPPS的应用程应遵循以下原则：

- 可运行于 CUPPS 所支持的操作系统上；
 - 必要组件的安装、配置和访问由应用自身完成；
 - 应用不会对平台构成安全风险；
 - 应用程序不能影响平台的稳定性；
- 应用程序不能违反任何支付卡行业数据安全标准。

4.2 逻辑系统结构

MH/T XXXX-XXXX的本部分只提供与CUPPS逻辑架构相关的信息，不提供具体的实现细节定义。CUPPS提供商可自由实现符合要求的CUPPS平台。CUPPS提供的统一逻辑架构见图1。

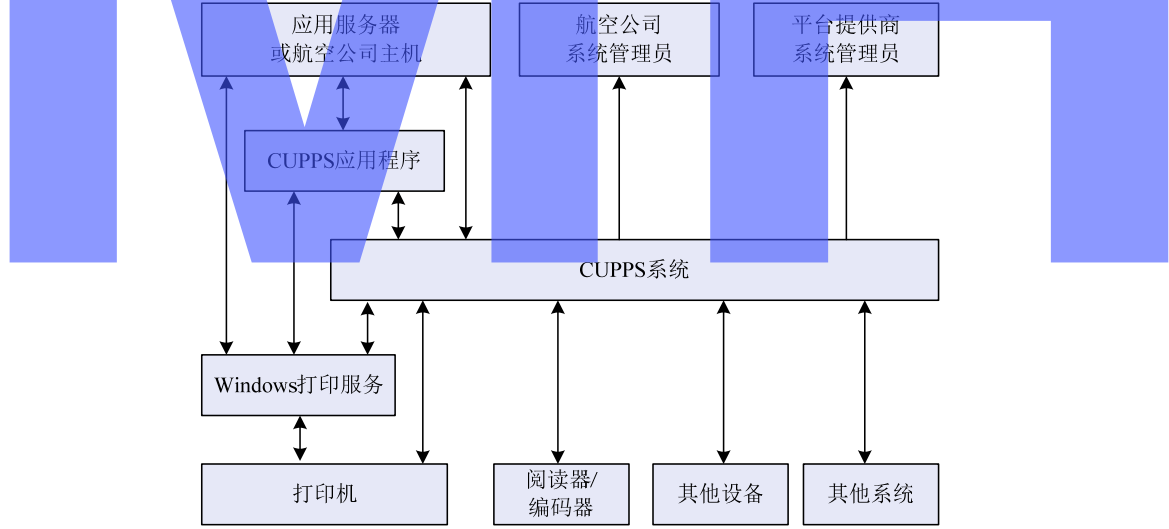


图1 CUPPS 系统逻辑架构

应用服务器或航空公司主机系统与CUPPS系统通过双向通信协议进行通信。CUPPS应用程序与CUPPS系统之间使用本部分定义的CUPPS平台接口进行通信。CUPPS应用程序只能通过CUPPS平台接口访问CUPPS管理的设备；CUPPS应用程序可以通过操作系统提供的基本服务，使用工作站提供的其他功能。航空公司系统管理员和平台提供商系统管理员通过CUPPS定义的管理接口与CUPPS平台通信。CUPPS平台通过事件报告与外部系统管理工具进行交互。

关于CUPPS更为详细的逻辑结构见图2。

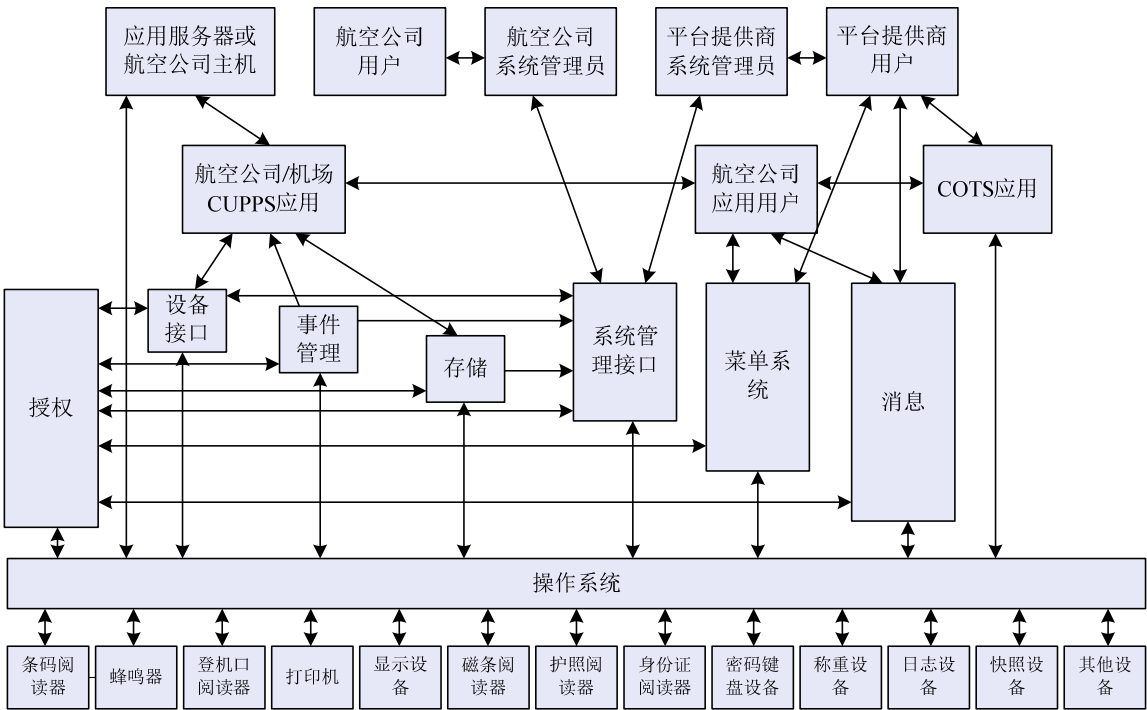


图2 CUPPS 系统逻辑结构

4.3 设备接口

CUPPS工作站支持的设备见表1。单个硬件设备可以支持本部分中描述的多个设备功能。CUPPS平台提供商可以自由选择符合要求的硬件设备。

5 CUPPS 系统功能

5.1 CUPPS 系统构成

CUPPS系统包括CUPPS平台、 CUPPS工作站、CUPPS用户、应用程序(CUPPS应用与COTS应用)、CUPPS设备的管理。这些对象之间的逻辑关联见图3。

CUPPS系统需要管理整个CUPPS平台，控制CUPPS工作站的启动和停止，并将CUPPS警报信息分发到CUPPS工作站。CUPPS工作站负责管理工作站的运行环境，CUPPS用户的启动和停止，将CUPPS警报信息传递给CUPPS用户。CUPPS用户负责管理特定CUPPS用户的运行环境，CUPPS应用程序和COTS 应用程序的启动和停止。应用程序负责管理特定CUPPS应用程序和COTS应用程序实例的运行环境。CUPPS设备负责对设备实例的逻辑关系的管理。

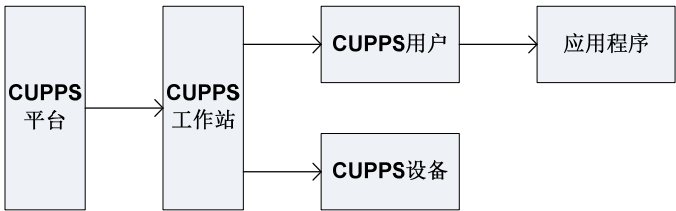


图3 CUPPS 系统构成

CUPPS系统各部分的构成满足以下要求：

- 每个 CUPPS 平台由零个或多个 CUPPS 工作站组成；
 - 每个 CUPPS 平台由零个或多个 CUPPS 设备组成；
 - 每个 CUPPS 工作站由零个或多个用户和 CUPPS 设备组成；
- 每个CUPPS用户由零个或多个应用程序组成。

表1 CUPPS 设备接口细节

缩写	名称	接口模式 ^{a)}				
		AEA(Association Europe Airlines) ^{b)}	标准	特殊	其他	Windows 打印服务
BC	条码阅读器	DNR ^{c)}	DR			
BE	蜂鸣器	DNR	DNR			
BG	登机口阅读器	DR	DR			
BP	登机牌打印机	DR				
BT	行李标签打印机	DR ^{d)}				
DD	显示设备	DR				
IR	身份证阅读器		DR			
MS	磁条阅读器		DR			
OC	护照阅读器		DR			
PP	密码键盘设备	DR				
PR	文档打印机		DR ^{d)}			DR
RW	其他设备				DR	
SD	称重设备	DNR	DNR			
SN	快照设备					
ZI	IATA 报文软件设备			DR		
ZL	日志软件设备			DR		

^{a)} 表中空白处表示NDNR，未定义也不强制要求实现的接口；另外，DR为定义并且必须实现的接口，DNR为定义但不强制要求实现的接口。

^{b)} 本部分中提到的AEA是指AEA 2009标准，其他版本的AEA不做要求。

^{c)} 在AEA模式下，BC是BG的一个子设备。

^{d)} 如果设备支持射频识别技术(Radio Frequency Identification, RFID)，则可包含 RFID 的功能。

5.2 命名标准

5.2.1 总体要求

CUPPS工作站名称、CUPPS设备名称应是标准化的，并且在全球站点实现唯一命名。该命名标准旨在使名称解析具备可靠性和可扩展性。

5.2.2 工作站和设备名称

工作站和设备命名标准支持全球唯一性标识，并且能够根据这个名称判断出工作站和设备的大概物理位置。

对于非测试环境下的CUPPS操作，平台应使用其所在的机场及城市代码。在测试环境下，可使用测试名称标志Z。无论什么情况下，平台提供商应保证工作站和设备名称的唯一性。

工作站或者设备的名称由字符串组成，并且需要包含以下信息：机场或者城市代码，具体的平台提供商代码，功能区代码，以及设备相关信息。

工作站或设备节点标准化命名的构成见表2。

表2 工作站或设备的名称构成

编号	长度	格式	内容
1	3	AN	机场代码，国际航协机场或者城市代码
2	1	AN	采用国际航协提供的平台提供商代码，见表 3
3	1	AN	航站楼编号，如果是测试节点则设为 Z
4	1~7	AN	平台提供商自定义可变字符，需包含表 4 中的功能区代码
5*	2	AN	表 1 中的设备类型码
6*	1	N	设备索引
注：*为可选项，AN为字母-数字，N为数字			

表3 平台提供商代码

代码	平台提供商
A	ARINC
B	IBS
C	Ink Aviation
D	BagDrop Systems
I	AirIT
K	IIAC and Asiana IDT
R	RESA
S	SITA
T	TravelSky
U	Ultra
X	任何新的平台提供商应到 IATA 的 CUWG 组注册才能获得提供商代码

功能区代码的描述见表4。

表4 功能区域代码

代码	描述	代码	描述
BG	行李提取	HH	手持设备
BH	行李搬运	JW	登机通道
BK	行李托运检查	LA	实验室
BO	办公室	LB	大厅
BS	行李扫描	LF	失物认领
Bn	地下楼层 n 为 1~9	LG	休息室
CK	值机柜台	LW	隔离厅
CM	海关	MD	主层

表 4 (续)

代码	描述	代码	描述
CN	引导处	Mn	夹层 n 为 1~9
CO	贵宾服务	RP	扶梯
CR	路边(临时停车区)	TD	签转柜台
CS	客户服务区	TK	售票区
DR	通道	TN	培训室
GT	登机口	TR	机场火车、地铁站
GH	传达室	nn	楼层 01~99

示例:

TAOT1CKE12 表示青岛流亭机场(TAO),TravelSky(T)提供 CUPPS 平台,1 号航站楼(1),E 区 12 号值机柜台(CKE12)工作站。TAOT1CKE12BP1 表示位于这个柜台上的 1 号登机牌打印机(BP1)。

5.3 CUPPS 平台功能

5.3.1 软件和硬件

5.3.1.1 操作系统

平台服务器可以使用任意平台提供商认为合适的操作系统,只要能够像基于微软windows的服务器一样提供以下功能并保证其实对于终端软件是透明的:

- 提供适当的访问控制机制共享网络和打印机;
- 终端用户认证基础结构要符合本部分的条款。

5.3.1.2 硬件

平台服务器没有预定义的硬件列表,其硬件配置由平台提供商决定。
服务器硬件可从任何硬件平台提供商获取。本部分不提供任何已授权、首选等类似的厂商推荐。

5.3.2 存储功能

5.3.2.1 应用存储

平台应提供一种机制存储 COTS 应用程序、CUPPS 应用程序及配置数据。应用存储可以基于服务器,也可基于工作站。

基于服务器存储应提供一个简单的单磁盘卷或一个复杂的磁盘阵列系统。底层存储机制对使用它的应用来说应是透明的。

平台应为每个航空公司提供最低限度 5 GB 的应用存储空间。

5.3.2.2 非应用存储

非应用存储包括数据、应用日志、临时文件等的存储。
平台应提供永久性全局存储、永久性局部存储和临时存储三种存储方式。
三种不同类型存储方式的特征见表5。

表5 三种存储方式的比较

特征	永久性全局存储	永久性局部存储	临时存储
持久性	无限	无限	应用程序使用期间
可由其他工作站访问	是	否	否
受航空公司保护	是	是	是
磁盘配额	1 GB	1 GB	1 GB
备份和恢复	可以	不可以	不可以
存储位置	网络或本地	本地	本地

5.3.2.3 永久性全局存储

永久性全局存储允许所有CUPPS工作站访问，应被站点数据备份策略覆盖。

CUPPS平台应提供永久全局存储供COTS应用程序和CUPPS应用程序使用。永久性全局存储设备应能被所有工作站访问。写入此存储区域的数据应无限期保留，其数据规模不能超过表5中所分配的配额。永久性全局存储区域应包含在安装备份和恢复策略中。

应用程序通过环境变量CUPPSGSD、CUPPSGSU和<authenticateResponse>消息中persistentGlobal变量值决定其永久存储位置。

5.3.2.4 永久性局部存储

永久性本地存储只允许CUPPS本地工作站访问，不应被站点数据备份策略覆盖。

CUPPS平台应提供永久性本地存储供应用使用。为确保最大吞吐量，永久性本地存储应部署在本地工作站上，其他工作站不能访问。写入此存储区域的数据应无限期保留，其数据规模不能超过表5中所分配的配额。应用程序通过环境变量CUPPSPLSD、CUPPSPLSU以及<authenticateResponse>消息中persistentLocal变量值决定其永久性本地存储位置。

5.3.2.5 临时存储

CUPPS平台应提供临时存储供应用使用。每个工作站的临时存储区域独立于其他工作站，一个工作站不允许访问其他工作站的临时存储区域。写入临时存储的数据应在应用程序终止时由平台自动删除。临时存储不必包含在安装备份和恢复策略中。

应用程序通过环境变量CUPPSSTD、CUPPSSTSU以及<authenticateResponse>消息中transient变量值决定其临时存储位置。

5.3.3 查询功能

5.3.3.1 平台应支持查询功能以便于应用程序获取有关运行环境的信息。查询功能应基于本部分中定义的全局命名标准。

5.3.3.2 平台查询功能应能确定哪些 CUPPS 管理设备配置可以使用。

5.3.3.3 平台可提供上述要求外的其他查询功能。

5.3.4 消息功能

5.3.4.1 CUPPS 平台应提供一定形式的广播消息。这类消息用于安全警报、管理员警报等。

5.3.4.2 CUPPS 平台应提供图形用户界面以输入要分发的 CUPPS 警报信息。图形用户界面应使用 CUPPS 菜单系统或其他安全机制，以保证只有 CUPPS 平台管理员才可访问该界面。

5.3.4.3 界面应显示以下信息：

- 管理员输入的消息内容；
- 输入的日期和时间；
- 发布消息的管理员信息。

5.3.4.4 平台不必对未在平台上运行的工作站实现消息队列功能，也不必提供浏览历史消息的功能。消息功能只为实现实时传递。

5.3.5 认证功能

5.3.5.1 平台应提供通用认证机制。在实现方式上，可使用平台的内置认证机制，也可借助操作系统的内置认证机制或者其他设施。

5.3.5.2 认证方案应提供基于用户名和密码的验证方式，并实现基于角色的身份验证和权限机制。

5.3.6 监控功能

平台应对关键对象实施监控。并且应给管理员提供一个图形用户界面来监控处于异常状态的对象，主要包括：

- 应用被要求运行但处于 aStp 状态；
- 应用处于 aZom 状态；
- 设备被要求启动但处于 dStp 状态；
- 设备处于 dErr 状态或 dZom 状态；
- 工作站被要求启动但处于 wStp 状态；
- 工作站处于 wZom 状态；
- 用户被要求启动但处于 uStp 状态；
- 用户处于 uZom 状态。

5.3.7 平台管理报告

5.3.7.1 总体要求

CUPPS 平台应提供合适的数据用于报告平台的可用性。

鉴于数据用途以及用户对报告需求的多样性，此处不定义报告的具体格式。CUPPS 平台应提供原始数据以供用户创建报告。报告可以使用各种工具完成，比如微软的 Excel 软件等。

平台不要求使用某种特定的方法获取这些数据，也不必为其提供某种特定的接口。

5.3.7.2 可用性报告

平台只需要报告对象处于异常状态的时段，主要包括：

- aZom 状态；
- dErr 状态；
- dZom 状态；
- pStp 状态；
- uZom 状态；
- wZom 状态。

平台报告的时段的最小间隔为 60 s，如果时段不是 60 s 的整数倍，将其取整到下一整数。

5.3.7.3 使用情况报告

CUPPS 平台应提供数据用于报告平台的使用情况。

平台应记录以下使用情况：

- 从用户的身份验证开始到用户完成注销的时间，精确到 60 s；
- 物理设备的每次使用情况。

5.3.8 设备管理界面

平台应为设备管理提供图形用户界面并实现以下功能：

- 设备管理，如安装新设备，移除现有设备等；
- 设备监控。
- 具体功能和实现方法由平台提供商决定。

5.3.9 日志功能

5.3.9.1 平台应实现对平台相关操作的综合日志记录功能，包括与应用交互的消息和数据流，与平台各模块交互的消息和数据流，与其他平台交互的消息和数据流。

5.3.9.2 平台应提供软件日志设备处理日志记录操作。

5.3.9.3 平台相关日志应遵循以下原则：

- 平台相关日记记录功能可被平台开启；
- 平台相关日志记录功能可被应用开启；
- 平台相关日记记录功能可根据系统的配置被强制禁止。

5.3.9.4 数据记录形式不能违背支付卡行业数据安全标准。

5.3.9.5 平台应给每个航空公司提供大小 1 GB 的永久性网络存储空间用于存储平台的相关日志。该存储区域可根据存储类型分区，如可执行文件、临时数据、应用日志等。平台日志至少保留 5 天。

5.3.10 令牌管理

5.3.10.1 令牌可被平台和应用程序用来执行各种任务，例如访问控制和数据安全性。

5.3.10.2 生成令牌的方法由平台决定，应用程序不能尝试从令牌值推断任何特定的信息。

5.3.10.3 在授权应用程序访问平台资源之前，平台应验证应用程序的每个实例。应用程序应给出 <authenticateRequest> 消息中要求的数值，平台可使用任意方法来执行认证。当应用程序成功通过验证后，平台授予其一个设备令牌。应用程序实例使用此令牌与平台进行所有后续连接。

5.3.10.4 当接口信息交互时，应用程序向平台提供明确的令牌。平台应先验证令牌再处理后续的交易。如果应用提供的令牌无效，交易会被拒绝并且标识相应的状态码。

5.3.10.5 当接口信息交互时，应用程序未向平台提供明确的令牌，平台应先验证打开逻辑会话时所用的令牌。如果应用提供的令牌无效，交易会被拒绝并且标识相应的状态码。

5.3.10.6 当 CUPPS 平台管理接口连接断开后，令牌立即失效。基于此过期令牌的后续交易都被拒绝并标识相应的状态码。

5.3.11 实例控制

5.3.11.1 CUPPS 平台应提供管理应用程序实例个数的功能。

5.3.11.2 应用程序运行实例个数的判断由底层操作系统的进程管理功能决定。

5.3.11.3 应用程序实例个数的管理功能基于应用程序配置。如果未配置，平台不限制应用程序的实例数。

5.3.11.4 如果平台阻止了应用程序的实例化，平台应记录<applicationInstBlockEvent>事件，显示相关应用程序、工作站和用户的信息。

5.3.12 标准环境变量

CUPPS平台应给应用程序提供一个用于描述环境的标准环境变量列表。

应用程序与平台之间信息交换所需的环境变量见表6。

表6 CUPPS 平台上运行时环境变量

变量	定义	例值
CUPPSACN	工作站所在的计算机的别名，由计算机、航空公司和用户构成，默认为 CUPPSCN	ABC1234
CUPPSAL2	航空公司的 2 字 IATA 代码或本地定义值	LH
CUPPSAL3	航空公司的 3 字 IATA 代码或本地定义值	DLH
CUPPSALA	IATA 结算代码	220
CUPPSAPD	应用路径，使用盘符和路径	Q: \APPS\LH\
CUPPSAPL	应用路径盘符	Q
CUPPSAPU	符合通用命名规则的应用路径	\\node\share\
CUPPSBNI	堡垒服务器 IP 地址	155.237.18.253
CUPPSBNP	堡垒服务器端口	1835
CUPPSCN	计算机名	LHRT4LB00302PR1
CUPPSDSP	默认 windows spooler 打印机	\\node\prtshare\
CUPPSGSD	永久性全局存储位置，盘符和路径	Z: \LH\
CUPPSGSL	永久性全局存储位置，盘符	Z
CUPPSGSU	永久性全局存储位置，符合通用命名规则	\\node\share\
CUPPSPLSD	永久性本地存储位置，盘符和路径	Z: \AA\
CUPPSPLSL	永久性本地存储位置，盘符	Z
CUPPSPLSU	永久性本地存储位置，符合通用命名规则	\\node\share\
CUPPSPLT	平台参数	Xxx
CUPPSPN	平台节点，节点名称用来连接以进行查询、日志等其他平台级别操作	LHRCUPPS
CUPPSPP	平台端口	7535
CUPPSTSD	临时存储位置，盘符和路径	Y: \A\B\C\
CUPPSTSL	临时存储位置，盘符	Y
CUPPSTSU	临时存储位置，符合通用命名规则	\\node\share\
CUPPSUN	用户名称，即当前会话的逻辑用户名，通常是航空公司名，也可为地服人员名称	LH
CUPPSXSD	XSD 位置，盘符和路径	W: \A\B\
CUPPSXSDL	XSD 位置，盘符	W
CUPPSXSU	XSD 位置，UNC	\\node\share\
CUPPSxxn	设备类型 xx 索引 n，例如，CUPPSPR1 表示第一个打印机，CUPPSOC1 表示第一个护照扫描器	LHRT4LB00302PR1

在标准环境变量中提供给应用程序的信息应与返回给应用程序的<authenticateResponse>消息中的信息保持一致。

除了在<authenticateResponse>消息中，平台将所有驱动器号和路径中的斜线转换为反斜线外，平台提供的标准环境变量值应遵循基本的数据类型规则，如长度限制、样式等。

5.3.13 任意环境变量

平台应支持配置任意的环境变量。

环境变量由任意工作站、航空公司和用户的组合在配置时定义。工作站、用户和应用的组合可提供一个键值以查找相应的值。

处理操作顺序如下：

- 为工作站定义变量，如果没有用户登录，则完成环境构建过程；
- 当用户在应用程序启动之前登录，首先为工作站定义变量，然后添加或者改写为用户定义的变量；
- 当应用程序启动之前，首先为工作站定义变量，然后添加或者改写为用户定义的变量、应用程序变量等，直到操作顺序完成。

5.3.14 IATA 报文接口

CUPPS 平台应提供一个标准的、公共的接口与非标准报文系统进行 IATA 报文交换。

IATA报文接口的模型见图4。

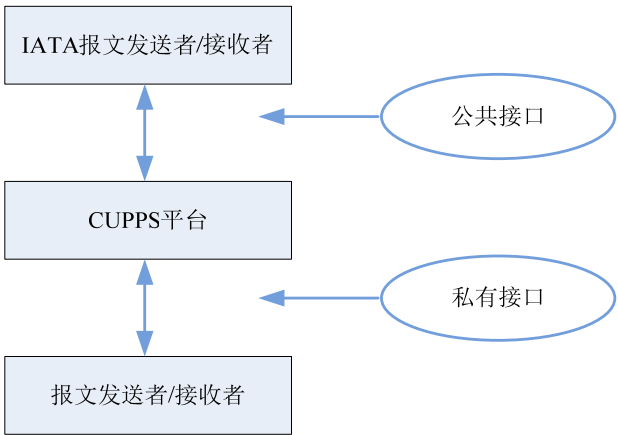


图4 IATA 信息接口

IATA报文发送者或接收者和非标准报文发送者或接收者可能需要集成引擎，本部分不要求CUPPS平台提供该集成引擎。

平台应提供IATA报文接口软件设备进行报文交互操作。

5.3.15 设备处理的可扩展性

CUPPS平台的设计应满足以下可扩展性需求：

- a) 支持至少 30 个设备：CUPPS 平台应为应用程序分配至少 30 个设备，以确保具有高需求的应用能获得足够的设备连接；
- b) 支持至少 254 个并发设备会话：CUPPS 平台应为逻辑设备提供至少 254 个并发连接，确保设备的访问具有足够的灵活性和并发性；

- c) 支持至少 4 个虚拟串口：CUPPS 工作站应提供至少 4 个虚拟串口，用于多种用途，如为其他串口设备提供连接。
- d) 串口编号至少支持到 COM255：CUPPS 平台支持的串口编号至少为 255，其中至少包含 4 个虚拟端口，以确保平台具有足够的灵活性来满足较高的连接性要求，而不必修改平台代码。串口编号应符合下述格式：COM1，COM2... COM255。

5.3.16 设备支持中文的要求

对于可打印输出内容的设备，例如BT、BP、PR，应包含符合中文GB18030要求的字库。

5.3.17 网络设置

平台应定制的网络设置的 Windows 注册表键值见表 7。

表7 Windows XP/Windows 7 网络参数

级别	设置	值
HKLM\SYSTEM\CurrentControlSet\Services\Tcpip\Parameters\Interfaces		
必须	EnablePMTUDiscovery	0x00000001
必须	EnablePMTUBHDetect	0x00000001
必须	KeepAliveInterval	0x00001388
必须	KeepAliveTime	0x00007530
推荐, XP 下	TcpMaxConnectRetransmissions	0x00000003
推荐	TcpMaxDataRetransmissions	0x00000005
推荐, XP 下	TcpWindowSize	0x0000CD50
推荐	MTU	0x0000058C
HKLM\SYSTEM\CurrentControlSet\Services\NetBT\Parameters\Interfaces		
推荐	SessionKeepAlive	0x0000AFC8

5.4 CUPPS 工作站功能

5.4.1 软硬件标准

5.4.1.1 操作系统

CUPPS工作站应满足以下要求：

- 操作系统要求为 32 位 Win7 Professional 版，最低要求是 Windows XP SP2 系统；
- 默认配置语言为简体中文，其他语言包或输入法可根据需要加载；
- 如果虚拟机或者类似技术(比如 VMWare 和 Citrix)能够提供一个与前述操作系统等效的运行环境系统(如对应用的内存分配等)，也可以使用。

5.4.1.2 硬件

CUPPS工作站包括的硬件见表8。

表8 平台工作站硬件

功能	描述
计算机	计算机及其内部组件
键盘	每个终端用户工作站应有一个标准的英文101+-的键盘
鼠标	每个工作站都应有一个，也可提供具有鼠标操作功能的键盘

平台工作站硬件的最低性能要求见表9。

表9 硬件性能标准

指标	最小值
硬盘大小	250 GB
内存	4 GB
CPU	双核2.0 GHz
网络带宽	100 Mbps

5.4.2 功能

CUPPS工作站应具有以下功能：

- a) 应用程序管理：CUPPS 工作站应实现应用管理功能；
- b) 身份验证：CUPPS 工作站应提供认证服务来验证用户是否有权限登录工作站或访问本地网络资源；
- c) COTS 软件互用性：COTS 软件应能够在 CUPPS 工作站上正常运行，并且不能对工作站或者当地网络带来明显的性能影响或安全风险。COTS 软件不能使用 CUPPS 系统的服务，比如设备等；
- d) CUPPS 应用要求：CUPPS 软件应能在 CUPPS 工作站上良好的运行，并且不能对工作站或者当地网络带来不合理的性能影响或者安全风险。每个 CUPPS 应用程序应遵循同一套命令标准，以便于查询应用程序版本或平台提供商；
- e) 事件管理：CUPPS 工作站应实现事件管理功能，以便于应用程序和系统之间使用管理接口进行事件信息的交换；
- f) 日志：CUPPS 工作站应提供一个用于应用程序支持和调试目的的通用日志接口；
- g) 菜单：CUPPS 工作站应提供具有安全性的菜单系统；
- h) 消息：CUPPS 工作站应提供消息传送功能，将管理人员发布的广播消息，在界面中展示给用户；
- i) 许可：CUPPS 工作站应提供功能实现对 CUPPS 工作站系统资源、网络资源、用户或航空公司数据的管理和安全访问；
- j) 工作站管理：CUPPS 工作站应提供管理功能以便于 CUPPS 系统管理、支持和使用；
- k) 存储：CUPPS 工作站应提供一个接口使应用程序能存储数据。接口应支持临时存储与永久存储；
- l) 全球统一命名原则：CUPPS 应使用全球统一的原则为 CUPPS 工作站及其设备进行命名；
- m) 性能：应为 CUPPS 工作站定义性能标准，包括登录过程时间、注销过程时间、通讯延迟时间和设备性能等。

5.5 COTS 软件互用

如果一个应用程序可以在工作站上运行，则其为COTS应用程序。如果应用程序因为依赖于CUPPS平台而不能运行，则其为CUPPS应用程序。

标准的COTS软件有.NET框架、Adobe Acrobat Reader、Microsoft Data Access Components、Microsoft Internet Explorer、VC6 libraries等。

6 平台管理

6.1 状态管理

6.1.1 总体要求

平台状态的定义见表10。

表10 平台状态摘要

状态	描述
pAlt	警报状态，表示平台向CUPPS工作站发布警报信息。只有当CUPPS平台管理员发布CUPPS警报信息时才会进入此状态
pStd	已启动状态，表示平台正在运行，平台的正常操作状态
pStg	正在启动状态，表示平台正在执行其启动引导操作
pStp	已停止状态，表示平台未运行，平台的默认状态
pSpg	正在停止状态，表示平台应执行必要操作来关闭其功能

平台管理状态机见图5。

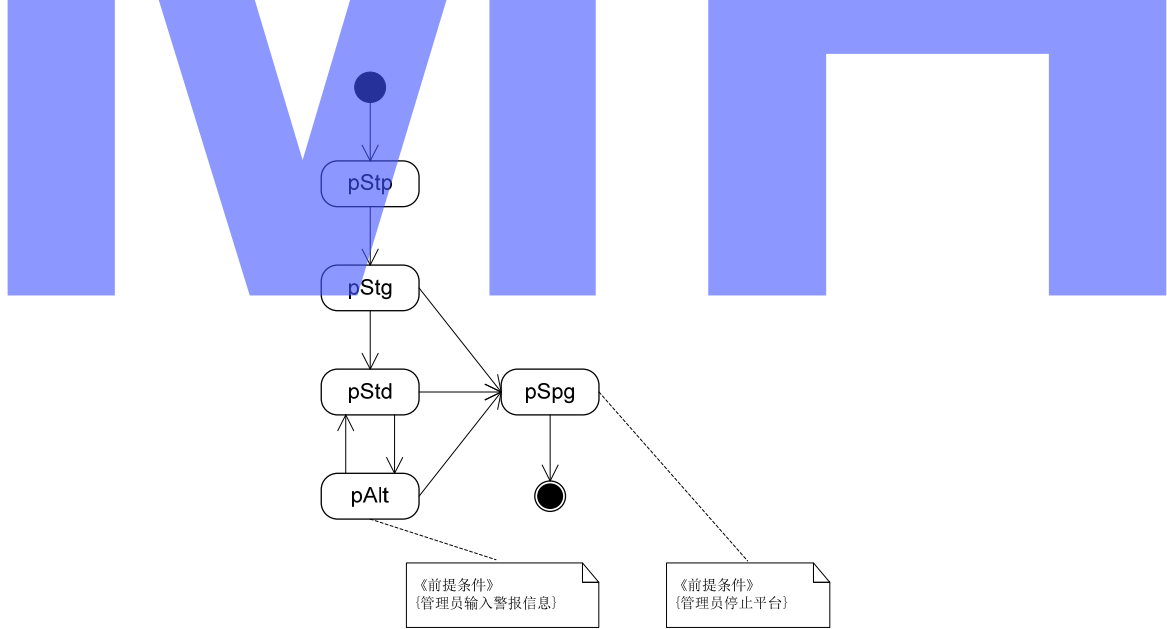


图5 平台状态机

平台状态机中允许的状态转换见表11。

表11 平台状态间的转换

出发状态	到达状态				
	pStp	pStg	pStd	pAlt	pSpg
pStp	——	√	—	—	—
pStg	—	—	√	—	√
pStd	—	—	——	√	√
pAlt	—	—	√	——	√
pSpg	√	—	—	—	——
注：“√”表示可以转换；“——”表示转换无意义；“—”表示不可能转换					

工作站管理状态机与远程状态机的交互见表12。

表12 平台状态机远程交互

状态	说明
pAlt	当平台有CUPPS警报信息发布时，平台通知所有的工作站
pSpg	当平台要停止时，平台停止所有工作站
pStd	当平台完成启动，会启动所有被设置为自启动的工作站
wAlt	当工作站传递CUPPS警报信息时，向平台报告传递过程
wSpg	当工作站要停止时，向平台报告
wStd	当工作站完成启动，向平台报告其成功完成启动
wZom	当工作站未能退出wSpg状态时，故障工作站向平台报告该事件

6.1.2 pAlt 状态

6.1.2.1 进入 pAlt 状态

当授权用户向平台发布CUPPS警报信息时，平台进入pAlt状态，触发<pAltEnteredEvent>事件。pAlt状态机见6。

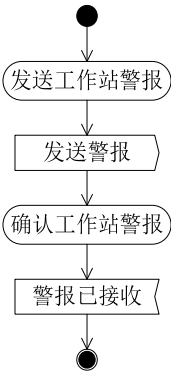


图6 pAlt 状态机

6.1.2.2 pAlt 状态处理过程

当进入pAlt状态后，平台将CUPPS警报信息的以下元素记录到日志中：
——输入警报信息的终端用户 ID；

——警报信息的内容。

为了监控CUPPS警报信息的分发过程，平台应执行以下操作：

- 记录一条日志消息表明 CUPPS 警报信息已经开始分发；
- 记录一条日志消息表明 CUPPS 警报信息已经开始分发给某个特定工作站；
- 记录一条日志消息表明 CUPPS 警报信息已经完成分发给某个特定工作站；
- 记录一条日志消息表明 CUPPS 警报信息已经完成分发。

平台会向所有处在wStd、wBsy或wAlt状态下的工作站发送CUPPS警报信息。警报信息的传输应是非阻塞的，这样向某一个工作站的警报传输不会妨碍向其他工作站的传输。

当有一个CUPPS警报信息传递给工作站后，这个警报会被添加到工作站的消息队列中，这时工作站马上进入wAlt状态。

工作站应主动确认警报信息是否正确收到，如果向某个工作站分发CUPPS警报信息失败，失败信息应记录到日志中。

6.1.2.3 退出 pAlt 状态

出现以下任意一种情况时，平台退出pAlt状态：

- 平台完成对所有适用工作站的 CUPPS 警报信息的发布；
- 平台管理员引导平台停止；
- 平台意外的出现故障。

当平台退出pAlt状态并且在进入下一个状态之前，触发<pAltExitedEvent>事件。

6.1.3 pSpg 状态

6.1.3.1 进入 pSpg 状态

当CUPPS平台管理员引导平台关闭时，平台进入pSpg状态，并记录一条消息说明平台已经停止。当平台进入pSpg状态后，触发<pSpgEnteredEvent>事件。
pSpg状态机见图7。

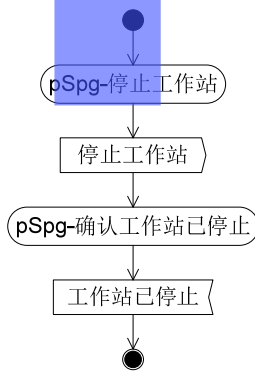


图7 pSpg 状态机

6.1.3.2 pSpg 状态处理过程

当平台进入pSpg状态后，执行以下操作：

- a) 平台在处理过程中以恰当方式记录以下日志：
 - 1) 进入 pSpg 状态；
 - 2) 停止平台的管理员信息；
 - 3) 开始执行关闭工作站的操作；

- 4) 开始关闭某个特定工作站;
- 5) 完成关闭某个特定工作站;
- 6) 完成停止信息的分发。
- b) 平台将所有工作站置为 wSpg 状态;
- c) 平台应向处于以下状态的工作站发送关闭命令:
 - 1) wStd 状态;
 - 2) wStg 状态;
 - 3) wBsy 状态;
 - 4) wAlt 状态;
 - 5) wSpg 状态。

当关闭工作站时, 关闭操作应是非阻塞的。

示例: 假设平台选择关闭 5 个工作站, 一台工作站的延迟关闭不能阻塞其他工作站的关闭。

6.1.3.3 退出 pSpg 状态

当平台完成关闭后退出 pSpg 状态。平台退出 pSpg 状态并且在进入下一个状态之前, 触发 <pSpgExitedEvent> 事件。

6.1.4 pStd 状态

6.1.4.1 进入 pStd 状态

当平台完成启动过程可以使用后, 进入 pStd 状态。当平台进入 pStd 状态时, 触发 <pStdEnteredEvent> 事件。pStd 状态机见图 8。

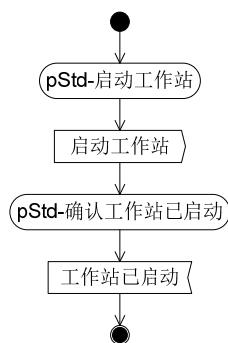


图8 pStd 状态机

6.1.4.2 pStd 状态处理过程

当平台进入 pStd 状态后, 启动所有配置为自启动的工作站。

6.1.4.3 退出 pStd 状态

出现以下任意一种情况时, 平台退出 pStd 状态:

- 平台要发布 CUPPS 警报信息;
- 平台被引导停止;
- 平台意外地出现故障。

当平台退出 pStd 状态并且在进入下一个状态之前, 触发 <pStdExitedEvent> 事件。

6.1.5 pStg 状态

6.1.5.1 进入 pStg 状态

当平台开始启动时进入pStg状态，触发<pStgEnteredEvent>事件。

6.1.5.2 pStg 状态处理过程

当平台进入pStg状态后，执行以下操作：

- 初始化日志系统；
- 在日志中记录平台正在启动；
- 将平台对象初始化为默认状态；
- 初始化平台及其对象的监控；
- 在日志中记录平台进入 pStg 状态。

6.1.5.3 退出 pStg 状态

当平台完成启动后退出pStg状态。

当平台退出pStg状态进入下一个状态之前，触发<pStgExitedEvent>事件。

6.1.6 pStp 状态

6.1.6.1 进入 pStp 状态

平台在以下任一情况下进入pStp状态：

- pStp 状态是平台的默认状态；
- 平台在正常关闭。

6.1.6.2 pStp 状态处理过程

平台在pStp状态时未运行，没有需处理的过程。

6.1.6.3 退出 pStp 状态

当平台开始启动时退出pStp状态。此时平台未运行，因此没有处理过程的要求。

6.2 平台信息工具

CUPPS平台应提供一个命令行工具输出平台实现和配置的关键信息。此工具提供给应用程序用于记录日志等。该工具的标准名称为CUPPSIT，位于系统路径PATH下。该工具可为一个可执行文件，也可使用CUPPSIT命令执行的脚本。

CUPPSIT工具应支持以下命令行参数：

- e：显示平台支持的加密算法；
- i：显示平台支持的接口标准；
- v：显示平台的版本号。

示例1：平台版本输出

```
1 C: > cuppsit -v
2
3 Platform...: myplatform
```

4 Version : 02.08

5

6 C: >

注：该示例中，平台名为myplatform，版本号为02.08。

示例2：平台加密算法输出

1 C: > cuppsit -e

2

3 Platform...: myplatform

4 Version : 02.08

5

6 Encryption : alg-a, alg-b

7 C: >

注 1：该示例中，平台名称为 myplatform，市场版本号为 02.08，支持的加密算法为 alg-a 和 alg-b。

注 2：当平台列出加密算法时，排在第一位的加密算法为默认算法。

示例3：平台接口信息

1 C: > cuppsit -i

2

3 Platform...: myplatform

4 Version : 02.08

5

6 IATA-CUPPS versions: 01.00, 01.01, 02.00

7 C: >

注：该示例中，平台名称为myplatform，版本号为02.08，支持的IATA CUPPS接口版本为01.00，01.01和02.00。

6.3 Windows 后台打印程序设置

平台应为应用程序提供Windows后台打印程序的访问。Windows后台打印程序的具体配置要求由本地站点决定。

7 工作站管理

7.1 状态管理

7.1.1 基本要求

工作站状态的定义见表13，工作站状态转换关系见图9。

状态转换表见表14。

工作站管理状态与其他管理状态机之间的相互作用见表15。

表13 工作站状态

状态	描述
wAlt	警报状态，表示平台向工作站发布一个 CUPPS 警报信息。该状态仅用于授权的管理员向工作站分发警报信息
wBsy	忙碌状态，表示工作站正在运行，且有终端用户登入工作站
wStd	已启动状态，表示工作站正在运行。该状态属于正常操作状态
wStg	正在启动状态，表示工作站正在执行引导程序
wStp	已停止状态，表示工作站没有运行。该状态是缺省状态
wSpg	正在停止状态，表示工作站正在执行停止操作
wZom	故障状态，表示工作站出现故障

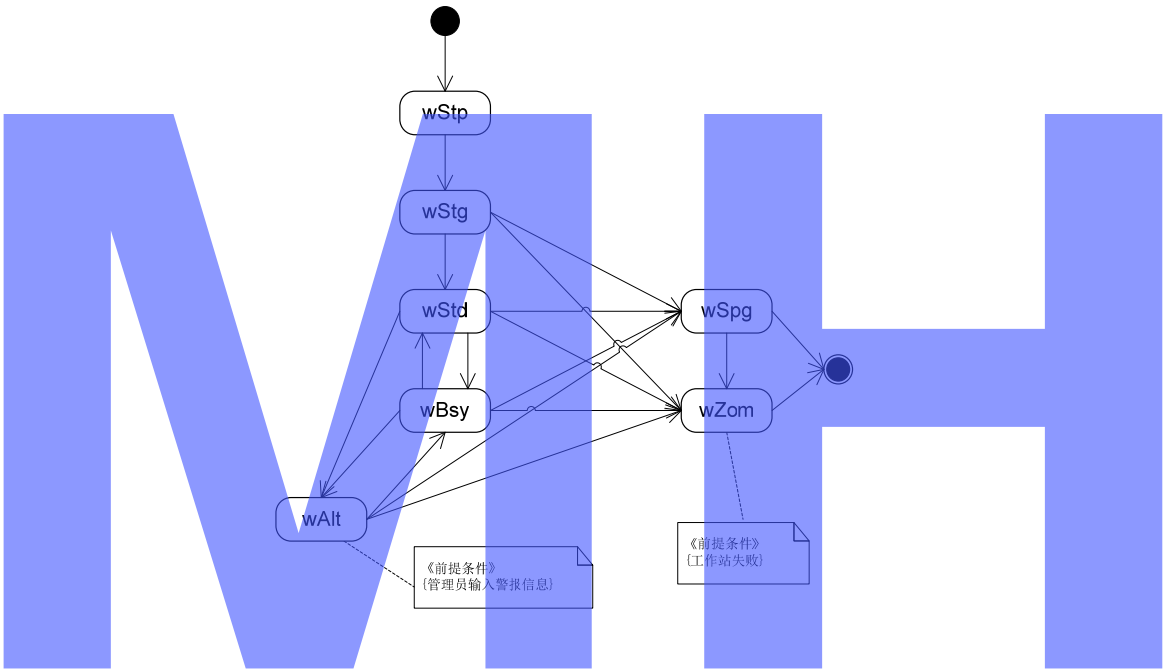


图9 工作站状态机

表14 工作站状态转换

出发状态	到达状态						
	wStp	wStg	wStd	wBsy	wAlt	wSpg	wZom
wStp	——	√	—	—	—	—	—
wStg	—	——	√	—	—	√	√
wStd	—	—	——	√	√	√	√
wBsy	—	—	√	——	√	√	√
wAlt	—	—	√	√	——	√	√
wSpg	√	—	—	—	—	——	√
wZom	√	—	—	—	—	—	——

注：“√”表示可以转换；“——”表示转换无意义；“—”表示不可能转换

表15 工作站状态机之间的影响

状态	描述
pStd	平台开启工作站。一种是自动触发，当平台启动时，启动配置为自启动的工作站；另一种是运行时事件，终端用户启动工作站
wStd	当工作站进入已启动状态，该工作站向平台报告状态的变化
pAlt	当平台要分配一个 CUPPS 警报信息，平台将使工作站从 wStd 或 wBsy 状态转入 wAlt 状态。如果工作站已经处于 wAlt 状态，工作站会保持此状态直到下一个警报被传送完成
wAlt	当工作站向终端用户显示警报信息，工作站要将该警报信息传送给平台
wSpg	当工作站进入停止状态，该工作站向平台报告状态的变化
wZom	当工作站退出超时，工作站向平台报告其状态变为 wZom 状态

7.1.2 wAlt 状态

7.1.2.1 进入 wAlt 状态

当CUPPS平台管理者输入一条展示给用户的CUPPS警报信息时，工作站进入wAlt状态，触发<wAltEnteredEvent>事件。

wAlt状态机见pAlt状态机。

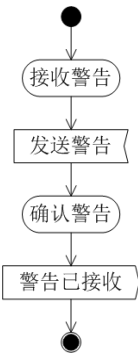


图10 wAlt 状态机

7.1.2.2 wAlt 过程

进入 wAlt 状态时，工作站应记录一条日志，表示进入 wAlt 状态。
为了把 CUPPS 警报信息传送给终端用户，平台应实现已定义的界面显示功能。

7.1.2.3 退出 wAlt 状态

从wAlt状态退出时，工作站应记录一条日志，表示退出wAlt状态。

7.1.3 wBsy 状态

7.1.3.1 进入 wBsy 状态

终端用户通过认证并执行应用程序时，工作站进入wBsy状态，触发<wBsyEnteredEvent>事件。wBsy状态机如pAlt状态机所示。

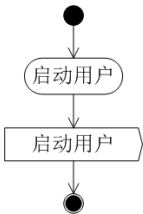


图11 wBsy 状态机

7.1.3.2 wBsy 过程

进入wBsy状态后，平台应执行以下操作：

- a) 记录进入 wBsy 状态的信息；
- b) 认证终端用户的 ID 和密码。如果终端用户不能在 10s 内通过认证，工作站退出 wBsy 状态，进入 wStd 状态；
- c) 通过平台内部认证机制验证终端用户。如果终端用户无法在 3 次内认证成功，记录登录失败信息、工作站 ID 和当时的日期时间。工作站离开 wBsy 状态并进入 wStd 状态。一旦用户通过认证，则工作站完成以下操作：
 - 1) 记录用户认证成功的消息；
 - 2) 启动所有配置为用户启动阶段的终端用户服务；
 - 3) 展示终端用户配置的的菜单系统。

7.1.3.3 退出 wBsy 状态

发生以下任何一种情况工作站退出wBsy状态：

- 工作站不再使用，返回 wStd 状态；
- 平台或者终端用户直接关闭工作站；
- 工作站遇到故障。

退出wBsy状态进入下一个状态之前，工作站触发<wBsyExitedEvent>事件。

退出wBsy状态时，平台应执行以下操作：

- 记录退出 wBsy 状态的消息；
- 如果终端用户从菜单系统选择注销功能，应执行以下操作：
 - 记录用户选择注销的消息；
 - 关闭用户服务；
 - 工作站环境重置为原始状态；
 - 工作站返回到 wStd 状态。

7.1.4 wSpq 状态

7.1.4.1 进入 wSpq 状态

通过平台或终端用户的操作，工作站进入wSpq状态，触发事件<wSpqEnterEvent>。wSpq状态机见图 12。

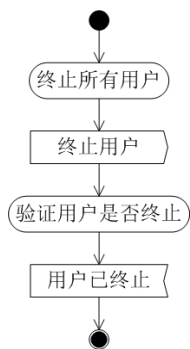


图12 wSpg 状态机

7.1.4.2 wSpg 过程

进入wSpg状态时平台应执行以下操作：

- a) 记录进入 wSpg 状态的消息；
- b) 停止所有终端用户。所有用户都离开当前状态进入 uSpg 状态。任何未能在 30 秒内正常停止的用户将会：
 - 1) 记录异常关闭的消息；
 - 2) 用户进入 uZom 状态并被回收。

用户被停止或者回收后，工作站离开wSpg状态进入wStp状态。
任何未能在30 s内进入wStp状态的工作站将进入wZom状态。
离开wSpg状态时，平台应记录离开wSpg状态的信息。

7.1.4.3 退出 wSpg 状态

发生以下任何一种情况，工作站都将离开wSpg状态：

- 工作站关闭程序成功执行；
- 工作站关闭程序意外失败。

退出wSpg状态进入下一个状态之前，工作站触发<wSpgExitedEvent>事件。

7.1.5 wStd 状态

7.1.5.1 进入 wStd 状态

在以下情况下工作站进入wStd状态：

- 工作站已经启动且处于可用状态；
- 工作站不再被使用而重新进入 wStd 状态。

工作站进入wStd状态时触发<wStdEnteredEvent>事件。wStd状态机见图13。

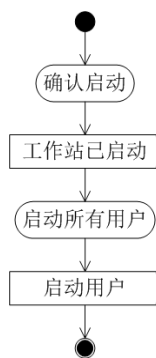


图13 wStd 状态机

7.1.5.2 wStd 过程

进入wStd状态时，平台应执行以下操作：

- 记录进入 wStd 状态的消息；
- 建立工作站运行环境；
- 工作站启动时，配置启动服务。

开启终端用户工作站服务时应执行如下操作：

- 使用工作站指定的环境变量设置工作站系统注册表；
- 设置工作站环境变量。

终端用户选择请求认证的菜单项时工作站离开wStd状态并进入wBsy状态。

7.1.5.3 退出 wStd 状态

发生以下任何一种情况时工作站退出wStd状态：

- 终端用户启动登录进程；
- 平台或者终端用户操作控制工作站停止工作；
- 工作站遇到故障。

退出wStd状态之后进入下一状态之前，工作站触发<wStdExitedEvent>事件。

7.1.6 wStg 状态

7.1.6.1 进入 wStg 状态

启动开启程序时工作站进入wStg状态并触发<wStgEnteredEvent>事件。

7.1.6.2 wStg 过程

进入wStg过程时，平台应执行以下操作：

- 记录进入 wStg 的消息；
- 建立工作站运行环境：
 - 建立所有网络驱动映像；
 - 开启所有 CUPPS 平台接口。

7.1.6.3 退出 wStg 状态

工作站因以下原因之一退出wStg状态：

- 工作站成功完成启动程序；

——工作站启动程序意外失败。
工作站退出wStg状态进入下一状态之前，触发<wStgExitedEvent>事件。

7.1.7 wStp 状态

7.1.7.1 进入 wStp 状态

wStp状态是工作站管理状态机的默认状态。默认状态下，平台初始化后所有工作站设置为wStp状态。
工作站进入wStp状态的几种情况如下：
——工作站被初始化为 wStp 状态；
——工作站正常关闭后进入 wStp 状态；
——故障工作站被回收时工作站进入 wStp 状态。
工作站进入wStp后，触发<wStpEnteredEvent>事件。
wStp状态机见图14所示。

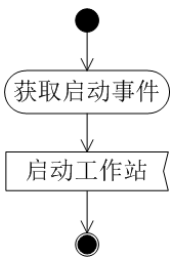


图14 wStp 状态机

7.1.7.2 wStp 过程

进入wStp状态时，平台记录进入wStp状态的消息。
一旦工作站停止工作，就不存在程序需求。
退出wStp状态时，平台记录一条提示退出的消息。

7.1.7.3 退出 wStp 状态

工作站退出wStp状态时，应记录退出wStp的消息。

7.1.8 wZom 状态

7.1.8.1 进入 wZom 状态

以下任何一种情况产生都会使工作站进入wZom状态：
——工作站在任何状态下意外失败；
——工作站未能在最大允许时间内完成关闭程序。

7.1.8.2 wZom 过程

进入wZom状态，平台应执行以下操作：
——记录进入 wZom 的信息；
——清理所有与工作站有关的平台资源。

7.1.8.3 退出 wZom 状态

当工作站被清理后退出wZom状态，返回到wStp状态。
退出wZom状态进入下一状态之前，工作站触发<wZomExitedEvent>事件。

7.2 工作站功能

7.2.1 警报图形用户界面

收到警报时，界面向用户展示警报、数据、发布时间，以及输入警报的管理员ID。应在最前端并以系统模式展示。如果收到多条警报，警报应以先进先出顺序调度。
用户应与系统模式UI交互后才被允许与系统其他UI交互。

7.2.2 日志

CUPPS工作站应综合记录工作站相关操作。包括应用程序的输入、输出、工作站的消息和数据流。
工作站记录相关规则：
——工作站相关记录功能可被平台开启；
——工作站相关记录功能可被应用程序开启；
——工作站相关日记记录功能可根据系统的配置被强制禁止。
当用户使用信用卡时，CUPPS管理设备会从磁条阅读器中读取信用卡信息，该信用卡信息最终会被传送给CUPPS应用。在这个过程中，记录在日志中的信用卡信息需要被特殊处理。
记录日志的软件设备逻辑名称定义为ZL。

8 用户管理

8.1 状态管理

8.1.1 基本要求

应用程序状态的定义见表16。表16中用户管理状态机中的每一个状态见图15。

表16 用户状态总结

状态	描述
uStg	正在停止状态，表示用户正在退出
uStd	已启动状态，表示用户正在运行。此状态为正常操作状态
uStp	已终止状态，表示用户没有登录。此状态为默认状态
uStg	正在启动状态，表示正在执行其启动引导操作
uSs	屏幕保护状态，表示正在显示屏幕保护程序
uTo	超时状态，表示用户在 600s 内没有活动。根据系统的配置，超时会导致用户变为 uSs 或者 uSpG 状态
uZom	故障状态，表示用户不能以期望的方式进行操作

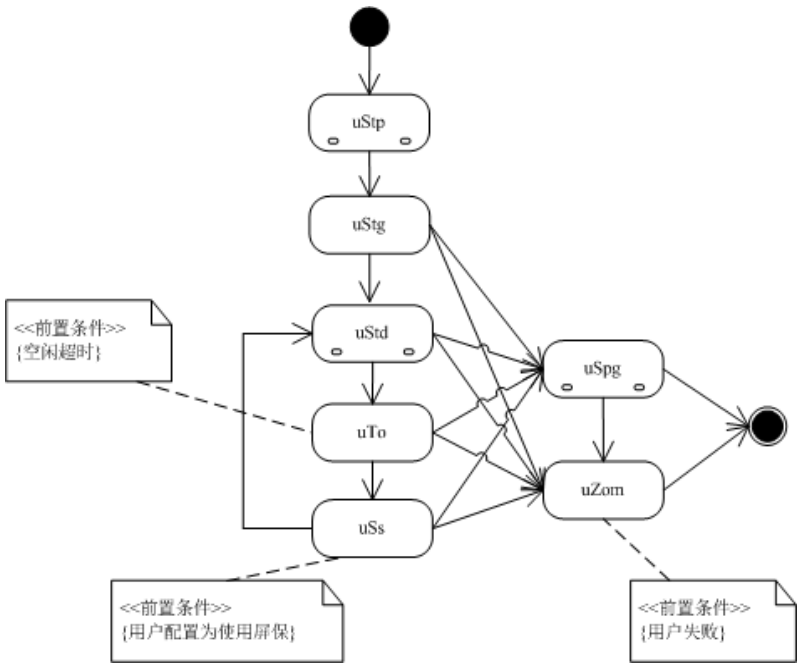


图15 用户管理状态机

表16中用户状态转换表见表17，该表能够从状态机的任意点进行向前或向后遍历。

表17 用户状态转换

初始状态	到达状态						
	uStp	uStg	uStd	uBsy	uAlt	uSpg	uZom
uStp	——	√	—	—	—	—	—
uStg	—	——	√	—	—	√	√
uStd	—	—	——	√	√	√	√
uBsy	—	—	—	——	√	√	√
uAlt	—	—	√	—	——	√	√
uSpg	√	—	—	—	—	——	√
uZom	√	—	—	—	—	—	——

注：“√”表示可以转换；“——”表示转换无意义；“—”表示不可能转换。

8.1.2 uSpg 状态

8.1.2.1 进入 uSpg

uSpg 状态表示用户在正常终止操作下的退出过程。通过平台或终端用户操作，用户进入 uSpg 状态，触发<uSpgEnterdeEvent>事件。

8.1.2.2 uSpg 过程

- 进入 uSpg，平台应执行下列操作：
- 记录一条进入 uSpg 的消息；
 - 终止所有终端用户的应用。应用程序退出当前状态进入 aSpg 状态。
- 任何未能在45 s内停止的应用将会：

- 记录一条表示错误退出的消息；
- 应用进入aZom状态，并被回收。

终端用户应用被终止或回收后，退出 aStd 状态进入 aSpg 状态。

任何未能在 45 s 内停止的终端用户服务将会：

- 记录一条表示错误退出的消息；
- 应用进入 aZom 状态，并被回收。

任何不能在 30 s 内进入 uStp 状态的用户将进入 uZom 状态。

退出 uSpg 状态，平台应记录一条消息表明退出 uSpg 状态。

8.1.2.3 退出 uSpg 状态

用户由于以下原因退出uSpg：

- 用户的退出程序成功完成；
- 用户的退出程序意外失败。

用户退出uSpg状态进入下一个状态前，触发<uSpgExitedEvent>事件。

8.1.3 uSs 状态

8.1.3.1 进入 uSs 状态

uSs状态表示工作站上的屏幕保护程序正在运行。当用户在600s内未进行操作时，进入uSs状态，触发<uSsEnteredEvent>事件。

8.1.3.2 uSs 过程

进入 uSs 状态，平台应执行以下操作：

- a) 记录一条表示进入 uSs 状态的消息；
- b) 显示平台指定的屏幕保护程序。

如果没有用户交互时，屏幕保护程序运行时间超过600 s, 用户退出uSs状态进入uSpg状态。

如果用户在600 s内进行了操作，提示输入用户名和密码用于验证。如果终端用户未在10 s内提交认证，屏幕保护程序会再次显示。屏保定时器不重置。如果用户认证次数超过3次，用户退出uSs状态进入uSpg状态。

终端用户认证由平台内部机制验证。如果终端用户认证失败次数超过 3 次，则：

- 记录一条登陆失败的消息，至少包含最近一次尝试的工作站 ID、日期和时间；
- 用户退出 uSs 状态进入 uStd 状态。

一旦通过验证，应记录一条消息，表明用户身份验证成功。

8.1.3.3 退出 uSs 状态

用户由于以下原因退出uSs状态：

- 用户在 600 s 内未进行操作。用户退出 uSs 状态进入 uSpg 状态；
- 用户意外失败。

用户退出uSs状态进入下一个状态之前，触发<uSsExitedEvent>事件。

8.1.4 uStd 状态

8.1.4.1 进入 uSs 状态

uStd是最主要的用户状态之一，表示用户已进行了完整的初始化。当用户已经启动并可以使用时，进入uStd状态。用户进入uStd状态后，触发<uStdEnteredEvent>事件。uStd状态的状态机见图16。

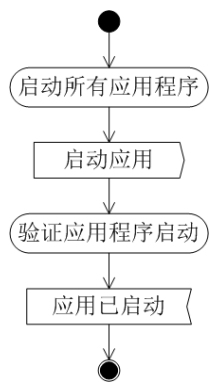


图16 uStd 状态机

8.1.4.2 uStd 过程

- 进入uStd状态，平台应执行以下操作：
- 记录一条进入 uStd 状态的消息；
 - 安装用户运行环境。
- 用户启动时，配置启动服务。
- 当启动终端用户服务时，应执行以下操作：
- 设置用户 Windows 注册表；
 - 设置用户环境变量，当用户在 600 s 内未进行操作，退出 uStd 状态进入 uTo 状态。

8.1.4.3 退出 uStd 状态

- 用户由于以下原因退出uStd状态：
- 平台或最终用户的行为导致用户停止；
 - 用户意外失败。
- 当用户超时计时器触发时，用户进入uTo状态。
- 用户在退出uStd状态进入下一个状态之前，触发<uStdExitedEvent>事件。

8.1.5 uStg 状态

8.1.5.1 进入 uStg 状态

- uStg状态表示正在启动，用户可进行初始化操作。uStg状态跟踪用户从启动开始到启动结束的过程。
- 当用户开始启动过程进入uStg状态时，应记录一条表示进入uStg状态的消息。
- 用户进入uStg状态，触发<uStgEnteredEvent>事件。

8.1.5.2 uStg 过程

- 当平台进入uStg状态时，应记录一条表示进入uStg状态的消息、建立用户运行环境，并执行以下操作：
- 建立网络驱动映射器；

——启动所有 CUPPS 平台接口。

8.1.5.3 退出 uStg 状态

以下原因会导致用户退出uStg状态：

——用户成功的完成启动程序；

——用户启动程序意外的失败；

用户在退出uStg状态进入下一个状态之前，触发<uStgExitedEvent>事件。

8.1.6 uStp 状态

8.1.6.1 进入 uStp 状态

uStp状态是用户管理状态机的默认状态。默认情况下，当平台初始化后，所有用户都将置于uStp状态。以下原因导致用户进入uStp状态：

——用户状态机初始状态为 uStp 状态；

——正确关机后用户进入 uStp 状态；

——故障用户被收回后，用户进入 uStp 状态。

当用户进入 uStp 状态时，触发<uStpEnteredEvent>事件。uStp 状态机见图 17。

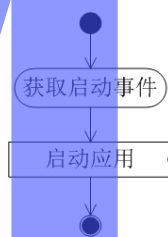


图17 uStp 状态机

8.1.6.2 uStp 状态过程

一旦进入uStp状态，平台应记录一条进入uStp状态的提示信息。

8.1.6.3 退出 uStp 状态

当用户开始启动时，退出uStp状态。

在用户退出uStp状态进入下一个状态之前，触发<uStpExitedEvent>事件。

8.1.7 uTo 状态

8.1.7.1 进入 uTo 状态

uTo状态表示用户因为没有操作而超时。当用户超时计时器触发时，用户进入uTo状态，触发<uToEnteredEvent>事件。

8.1.7.2 uTo 状态过程

进入uTo状态，平台将执行以下操作：

——记录一条进入 uTo 状态的信息；

——如果用户配置使用屏幕保护程序，则退出 uTo 状态进入 uSs 状态；

——如果用户配置不使用屏幕保护程序，则退出 uTo 状态进入 uSpg 状态。

8.1.7.3 退出 uTo 状态

以下原因导致用户退出 uTo 状态：

- 如果用户配置使用屏幕保护程序，则退出 uTo 状态进入 uSs 状态；
- 如果用户配置不使用屏幕保护程序，则退出 uTo 状态进入 uSpg 状态；
- 用户意外失败。

在用户退出uTo状态进入下一个状态之前，触发<uToExitedEvent>事件。

8.1.8 uZom 状态

8.1.8.1 进入 uZom 状态

以下原因导致用户进入uZom状态：

- 用户在其他状态下意外失败；
 - 用户未在允许的最大时间内完成关闭操作。
- 当用户进入uZom状态时，触发<uZomEnteredEvent>事件。

8.1.8.2 uZom 过程

进入uZom状态，平台执行以下动作：

- 记录进入 uZom 状态的提示信息；
- 清除与用户有关的所有平台资源。

8.1.8.3 退出 uZom 状态

用户被回收时，退出uZom状态，返回uStp状态。

当用户退出uZom状态进入下一个状态之前，触发<uZomExitedEvent>事件。

8.2 用户功能

8.2.1 菜单

CUPPS平台应根据用户的角色构建不同的菜单，展示给终端用户，并根据用户的行为执行相应的操作。

CUPPS平台在满足以下要求时可自由实现菜单功能：

- 菜单应只显示当前用户允许执行的菜单项目；
- 菜单项应考虑应用程序和工作站的功能区代码，符合功能区环境。假设一个用户登录了值机功能区的工作站，给这个用户显示的菜单项不能包含不允许执行的菜单项，或非值机功能区定义的菜单项。

8.2.2 日志

CUPPS平台要对用户的操作，实现全面的日志记录功能。该记录包括应用程序的输入、输出；用户组件的消息和数据流。

用户相关记录符合以下原则：

- 用户相关记录可被平台打开；
- 用户相关记录可被应用程序打开；
- 通过系统配置，用户相关记录可以被关闭，忽略其他平台和应用程序的开启要求。

9 应用程序管理

9.1 状态管理

9.1.1 基本要求

应用程序的状态定义见表18。

表18 应用状态总结

状态	描述
aAth	认证状态，表示应用在超时之前获得认证。如果超时，应用被停止
aCts	COTS状态，表示运行的应用是一个COTS应用
aSpg	正在停止状态，表示应用正在停止
aStd	已启动状态，表示一个应用正在运行并且已认证
aStg	正在启动状态，表示应用正在执行其启动操作
aStp	已停止状态，表示应用不再运行。这是一个默认的状态
aZom	故障状态，表示应用崩溃或不能以期望的方式运行

应用程序管理状态机见图18。

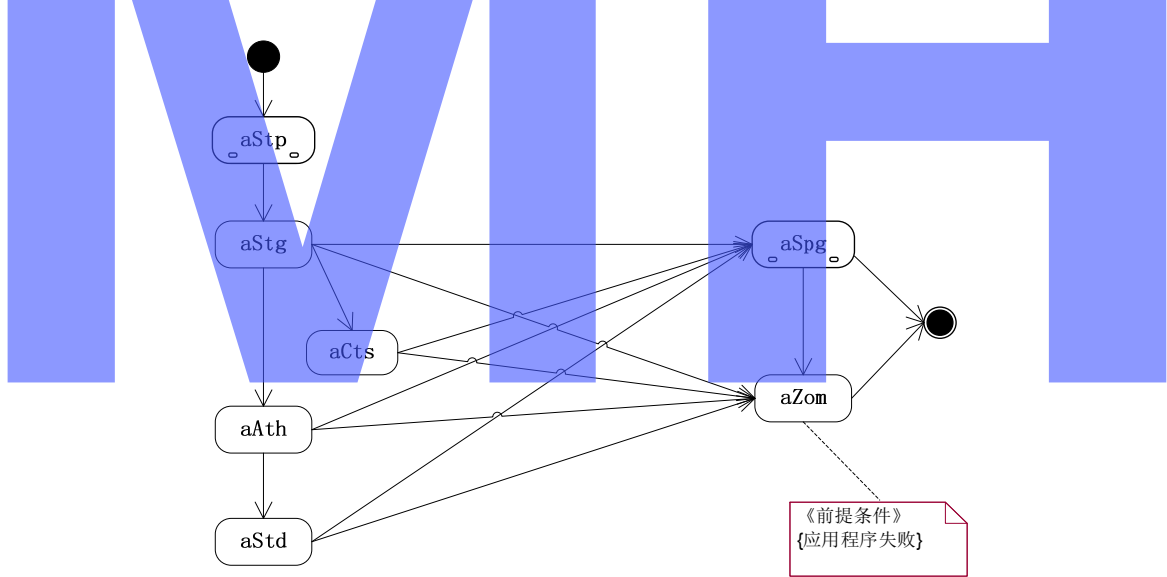


图18 应用管理状态机

应用程序管理各状态的转换见表19。

应用程序管理状态机与其他管理状态机之间的相互作用见表20。

表19 户状态转换

目的状态	初始状态						
	aStp	aStg	aCts	aAth	aStd	aStg	aZom
aStp	——	√	—	—	—	—	—
aStg	—	——	√	√	—	√	√
aCts	—	—	——	—	—	√	√
aAth	—	—	—	——	√	√	√
aStd	—	—	—	—	——	√	√
aStg	√	—	—	—	—	——	√
aZom	√	—	—	—	—	—	——
注：“√”表示可以转换；“——”表示转换无意义；“—”表示不可能转换							

表20 应用状态机之间的相互作用

状态	描述
uStd	应用根据用户启动而自动启动
aCts	平台向相关用户报告COTS应用成功启动
aStd	平台向相关用户报告CUPPS应用成功启动
uSpg	用户的关闭过程会触发属于该用户的应用关闭
aSpg	平台向有关用户报告应用的终止
aZom	平台向有关用户报告失败的应用

9.1.2 aAth 状态

9.1.2.1 进入 aAth 状态

aAth状态表示应用程序已经启动和初始化，但还没有通过认证。当应用程序完成启动时，从aStg状态进入aAth状态，触发<aAthEnteredEvent>事件。

9.1.2.2 aAth 过程

进入aAth状态，平台应等待应用程序的身份验证，且应在5 s内回应应用程序的身份验证请求。

9.1.2.3 退出 aAth 状态

发生以下任何一种情况，应用程序退出aAth状态：

- 应用程序验证失败，应用程序转入 aSpg 状态；
- 应用程序由于平台或终端用户的操作直接停止，应用程序转入 aSpg 状态；
- 应用程序意外失败，应用程序转入 aZom 状态；
- 应用程序成功验证，应用程序转入 aStd 状态。

应用程序退出aAth状态进入下一个状态之前，触发<aAthExitedEvent>事件。

9.1.3 aCts 状态

9.1.3.1 进入 aCts 状态

aCts状态表示COTS应用程序在平台上运行。对于配置为COTS的应用程序，一旦完成其启动，就转入aCts状态。

当应用程序进入aCts状态，触发<aCtsEnteredEvent>事件。

9.1.3.2 aCts 状态过程

进入aCts状态，平台应记录一条消息，表明进入aCts状态。

9.1.3.3 退出 aCts 状态

发生以下任何一种情况，应用程序退出aCts状态：

- 应用程序不再使用，申请进入 aSpg 的状态；
- 应用程序由于平台或终端用户的操作直接停止；
- 应用程序意外失败。

根据定义，只有CUPPS应用程序才可要求平台服务。如果COTS应用程序申请CUPPS平台服务，平台将记录一条消息，触发<cotsApplicationInterfaceEvent>事件，应用程序转入aSpg状态。

当应用程序退出aCts状态进入下一个状态之前，触发<aCtsExitedEvent>事件。

9.1.4 aSpg 状态

9.1.4.1 进入 aSpg 状态

aSpg状态表示应用正在执行退出操作。aSpg状态跟踪应用退出开始到退出结束的过程。发生以下任何一种情况，应用程序进入aSpg状态：

- 应用程序通过<byeRequest>信息给平台发退出信息；
- 平台通过给应用程序发送<applicationStopCommandRequest>引导其退出。

应用程序进入aSpg状态，触发<aSpgEnteredEvent>事件。aSpg状态机见图19。



图19 aSpg 状态机

9.1.4.2 aSpg 状态过程

进入aSpg状态，平台应执行下列操作：

- 记录一个信息，提示进入 aSpg 状态。
- 所有终端用户应用程序应终止。应用程序离开当前状态并且进入 aSpg 状态。

任何应用程序未能在 45 s 内正确停止将执行以下操作：

- 记录一条消息，指示不正确的退出；
- 应用程序进入 aZom 状态并且被清除。
- 任何未在 45 s 内进入 aStp 状态的应用程序，进入 aZom 状态；
- 平台给 CUPPS 应用程序发送一条终止消息。

终端用户应用程序和服务都终止后，应用程序释放占用的以下平台资源：

- 释放所有被锁定的设备；
- 取消所有待处理的作业。

9.1.4.3 退出 aSpg 状态

发生以下任何一种情况，应用程序退出aSpg状态：

- 应用程序成功完成关闭程序。如果应用程序在 45 s 内关闭，则进入 aStp 状态；
- 应用程序的关闭程序意外失败。如果应用程序没有在 45 s 内关闭，则进入 aZom 状态。

应用程序退出aSpg状态进入下一个状态之前，触发<aSpgExitedEvent>事件。

9.1.5 aStd 状态

9.1.5.1 进入 aStd 状态

当应用程序开始启动程序，进入 aStd 状态。应用程序进入 aStd 状态后触发<aStdEnteredEvent>事件。

9.1.5.2 aStd 状态过程

进入 aStd 状态，平台应记录一条信息，表示进入 aStd 状态。

9.1.5.3 退出 aStd 状态

发生以下任何一种情况，应用程序退出aStd状态：

- 应用程序开始认证过程，退出 aStd 状态；
- 平台或终端用户直接命令应用程序停止；
- 应用程序意外失败。

当应用程序退出aStd状态进入下一个状态之前，触发<aStdExitedEvent>事件。

9.1.6 aStg 状态

9.1.6.1 进入 aStg 状态

当应用程序开始启动程序，进入aStg状态。应用程序进入aStg状态后触发<aStgEnteredEvent>事件。

9.1.6.2 aStg 状态过程

一旦进入aStg状态，平台应执行如下操作：

- 记录一个日志，提示进入 aStg 状态；
- 根据应用程序配置，设置 Windows 注册表；
- 设置应用程序的环境变量；
- 启动配置为应用程序启动阶段的服务。

9.1.6.3 退出 aStg 状态

发生以下任何一种情况，应用程序退出aStg状态：

- 应用程序成功完成启动；
- 应用程序启动意外失败。

当应用程序退出aStg状态进入下一个状态之前，触发<aStgExitedEvent>事件。

9.1.7 aStp 状态

9.1.7.1 进入 aStp 状态

发生以下任何一种情况，应用程序进入aStp状态：

- 应用程序的初始状态；
 - 应用程序正确关闭；
 - 故障应用程序被清除。
- 应用程序进入aStp状态后触发<aStpEnteredEvent>事件。
aStp状态机见图20。

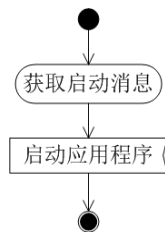


图20 aStp 状态机

9.1.7.2 aStp 状态过程

进入aStp状态，平台应记录一条信息，提示进入aStp状态。
所有aStp状态程序应在30 s内完成，否则平台将应用程序转入aZom状态。

9.1.7.3 退出 aStp 状态

当应用程序开始启动，退出aStp状态。
当应用程序退出aStp状态进入下一个状态之前，触发<aStpExitedEvent>事件。

9.1.8 aZom 状态

9.1.8.1 进入 aZom 状态

发生以下任何一种情况，应用程序进入aZom状态：

- 应用程序在其他状态下意外失败；
- 应用程序在允许的最大时间内没有完成关闭程序。

应用程序进入aZom状态后触发<aZomEnteredEvent>事件。

9.1.8.2 aZom 状态过程

进入aZom状态，平台应执行以下操作：

- 记录一个信息，提示进入 aZom 状态；
- 平台使用一种机制，如底层操作系统强行终止应用程序；
- 清除与应用程序有关的所有平台资源。

9.1.8.3 退出 aZom 状态

当应用程序被回收后退出aZom状态。一旦被回收，应用程序返回aStp状态。
当应用程序退出aZom状态进入下一个状态之前，触发<aZomExitedEvent>事件。

9.2 日志功能

CUPPS平台需要实现对应用程序相关的操作的全面的日志记录功能。记录包括传入、传出应用程序或平台应用组件的信息流和数据。

应用程序相关记录遵循以下原则：

- 平台可打开应用程序相关记录；
- 应用程序可打开应用程序相关记录；
- 通过系统的配置，应用程序相关记录可以被阻止，并忽略其他平台和应用程序的开启要求。

10 设备管理

10.1 接口模式

10.1.1 基本要求

CUPPS提供了多种设备接口模式。每种模式具有一个逻辑命令集、数据格式、隐式状态机等。

10.1.2 标准模式

标准模式是CUPPS定义的基本操作模式。标准模式严格执行设备管理功能和接口定义。
标准模式适用于票证文件或非票证文件。

10.1.3 AEA 模式

AEA模式是由欧洲航空公司协会定义的AEA2009标准命令和数据格式规范。
MH/T XXXX的本部分定义与AEA2009标准规范相关的CUPPS平台接口和功能需求。更多关于AEA的信息，请参考AEA2009标准。
AEA模式适用于票证文件或非票证文件。

10.1.4 其他模式

其他模式是一种访问设备自身提供 **API** 的机制。在其他模式下，应用程序自己负责设备接口维护和操作。除非底层设备的 **API** 提供了必要功能，其他模式不适用于票证文件。

10.2 状态管理

10.2.1 基本要求

设备状态的定义见表 21。

表21 设备状态

状态	描述
dBsy	繁忙状态，表示设备被锁定和使用
dErr	错误状态，表示设备有故障或类似异常状态
dLkd	锁定状态，表示设备被应用锁定
dSpg	正在终止状态，表示设备正在终止
dStd	已启动状态，表示设备在运行
dStg	正在启动状态，表示设备在运行它的启动操作
dStp	已终止状态，表示设备不在运行。这是一个默认状态
dZom	故障状态，表示设备崩溃或不能以期望的方式运行

设备管理状态机见图21。

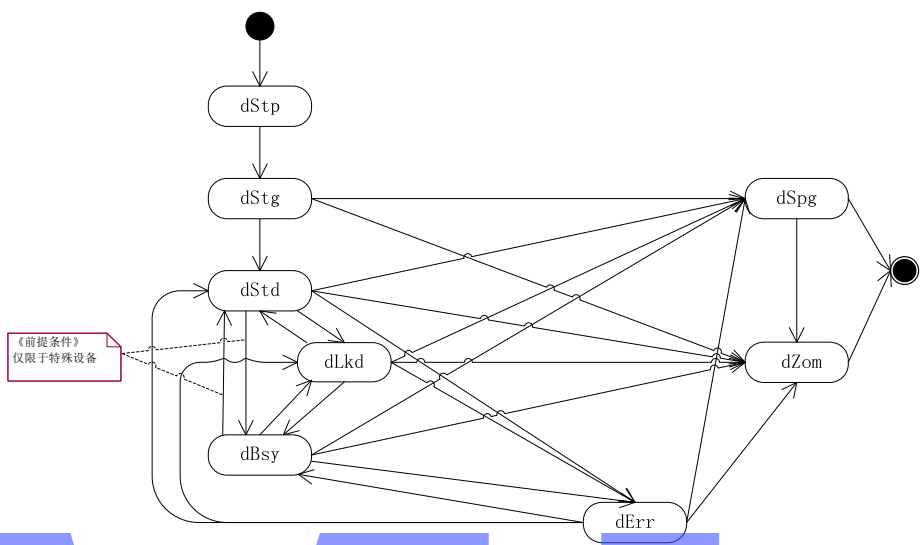


图21 设备管理状态机

允许的状态转换见表22，该表能够从状态机的任意点进行向前或向后遍历。

表22 设备状态转换

初始状态	目的状态							
	dStp	dStg	dStd	dLkd	dBsy	dErr	dSpg	dZom
dStp	——	√	—	—	—	—	—	—
dStg	—	——	√	—	—	—	√	√
dStd	—	—	——	√	—	√	√	√
dLkd	—	—	√	——	√	√	√	√
dBsy	—	—	—	√	——	√	√	√
dErr	—	—	√	√	√	——	√	√
dSpg	√	—	—	—	—	—	——	√
dZom	√	—	—	—	—	—	—	——

注：“√”表示可以转换；“——”表示转换无意义；“—”表示不可能转换。

一旦应用程序成功与设备连接，平台向每个应用程序发送状态转换事件。这与应用程序订阅每个设备的状态转换通知具有同样的效果。

如果通用事件订阅功能超出了范围，平台仍要求向已经获得连接设备的应用程序发送状态转变事件。

平台需要编写日志来记录每个设备状态的进入和退出。

10.2.2 dBsy 状态

10.2.2.1 进入 dBsy 状态

dBsy状态与dStd状态是设备的两个最主要状态。dBsy状态代表设备正在平台上使用。设备的启动、退出等相关程序因平台而异，因此本部分没有定义。

平台需要监视设备在dLkd状态下花费的时间，如果锁定的时间超过60 s，锁定被解除并返回dStd状态。当设备进入dBsy状态，跟踪锁定时间的定时器停止。当设备离开dBsy状态，计时器重新启动，进入dBsy状态。

当平台命令设备执行一个特定的任务时，设备进入dBsy状态。

当设备进入dBsy状态，触发<dBsyEnteredEvent>事件。

10.2.2.2 dBsy 状态过程

进入 dBsy 状态，平台应执行以下操作：

- 记录一个信息，提示进入 dBsy 状态；
- 停止与当前锁定时间有关的计时器；
- 设备进行 I/O 操作。

10.2.2.3 退出 dBsy 状态

发生以下任意一种情况，设备离开 dBsy 状态：

- 设备完成操作，返回 dLkd 状态。
- 设备被命令停止，进入 dSpg 状态。
- 设备出现故障，进入 dErr 状态。

当设备退出 dBsy 状态进入下一个状态之前，触发<dBsyExitedEvent>事件。

退出 dBsy 状态后，平台应执行以下操作：

- 重置和重新启动与当前锁定时间有关的定时器；
- 记录一条消息，表明离开 dBsy 状态；
- 通知所有已经获得设备连接的应用程序。

10.2.3 dErr 状态

10.2.3.1 进入 dErr 状态

当平台检测设备运行不正常时，设备进入 dErr 状态。

10.2.3.2 dErr 状态过程

进入 dErr 状态，平台应执行以下操作：

- 记录一条信息，提示进入 dErr 状态；
- 释放设备上的所有锁定；
- 把 dErr 状态通知给连接的所有应用程序。

退出 dErr 状态，平台应记录一条信息，提示离开 dErr 状态。

10.2.3.3 退出 dErr 状态

发生以下任何一种情况，设备退出dErr状态：

- 设备错误被排除。如果设备被锁定，则返回 dLkd 状态。如果没有锁定，则返回 dStd 状态；
- 平台或管理员命令该设备停止操作；
- 设备意外失败。

当设备退出 dErr 状态进入下一个状态之前触发<dErrExitedEvent>事件。

10.2.4 dLkd 状态

10.2.4.1 进入 dLkd 状态

dLkd状态表示一个特定设备被单一应用程序锁定。

发生以下任何一种情况，设备进入dLkd状态：

- 当应用程序申请锁定，其他应用程序未对其锁定。
- 当锁定的设备离开 dErr 状态，返回 dLkd 状态。

当设备进入 dLkd 状态，触发<dLkdEnteredEvent>事件。

10.2.4.2 dLkd 状态过程

进入 dLkd 状态，平台应执行以下操作：

- 记录一个信息，提示进入 dLkd 状态；
- 对于输出设备如打印机、显示器等，阻止所有其他的写操作；
- 对于输入设备如读卡器、护照阅读器等，阻止所有其他的读操作；
- 启动一个定时器，跟踪锁定持续时间；
- 刷新与设备关联的缓冲区。

10.2.4.3 退出 dLkd 状态

发生以下任何一种情况，设备退出 dLkd 状态：

- 设备不再使用，设备进入 dStd 状态；
- 平台命令设备停止，设备进入 dSpg 状态；
- 设备失败，进入 dZom 状态或 dErr 状态；
- 锁定时间超过 60s 没有 I/O 操作，进入 dStd 状态。

当设备退出 dLkd 状态在进入下一个状态之前，触发<dLkdExitedEvent>事件。

退出 dLkd 状态，平台应执行以下操作：

- 记录一个信息，提示离开 dLkd 状态；
- 刷新与设备关联的缓冲区。

10.2.5 dSpg 状态

10.2.5.1 进入 dSpg 状态

dSpg 状态表示关闭程序，设备使用正确的方式终止其操作。dSpg 状态跟踪设备的关机程序，直到设备停止。当平台或管理员引导设备停止时，设备进入 dSpg 状态。设备进入 dSpg 状态后触发<dSpgEnteredEvent>事件。

10.2.5.2 dSpg 状态过程

进入 dSpg 状态，平台应执行以下操作：

- 物理 I/O 端口应得到释放；
- 记录一条信息，表明进入 dSpg 状态；
- 平台应取消设备接口队列中等待的所有工作；
- 平台应通知所有与设备具有活跃连接的终端用户应用程序；
- 平台应让所有设备凭证无效；
- 记录凭证失效与发给应用程序的通知信息。

退出 dSpg 状态后，平台应记录一个信息，提示离开 dSpg 状态。

10.2.5.3 退出 dSpg 状态

发生以下任何一种情况，设备退出 dSpg 状态：

- 设备关闭程序成功完成；
- 设备关闭程序意外失败。

当设备退出 dSpg 状态进入下一个状态之前，触发<dSpgExitedEvent>事件。

10.2.6 dStd 状态

10.2.6.1 进入 dStd 状态

dStd 状态表示设备已经完全初始化。发生以下任何一种情况，设备进入 dStd 状态：

- 设备启动并且可以被使用；
- 设备不再被锁定重新进入 dStd 状态。

设备进入 dStd 状态后触发<dStdEnteredEvent>事件。

10.2.6.2 dStd 状态过程

进入 dStd 状态后，平台应记录一个信息，提示进入 dStd 状态：

退出 dStd 状态后，平台应记录一个信息，提示离开 dStd 状态。

10.2.6.3 退出 dStd 状态

发生以下任何一种情况，设备退出 dStd 状态：

- 平台或管理员命令设备停止；
- 设备意外失败。

当设备退出 dStd 状态进入下一个状态之前，触发<dStdExitedEvent>事件。

10.2.7 dStg 状态

10.2.7.1 进入 dStg 状态

dStg 状态表示设备使用启动程序正确初始化的状态。dStg 状态跟踪设备从开始启动程序到设备启动完成的过程。当设备开始启动程序就进入 dStg 状态，且触发<dStgEnteredEvent>事件。

10.2.7.2 dStg 状态过程

一旦进入 dStg 状态，平台应记录一个信息，提示进入 dStg 状态。

10.2.7.3 退出 dStg 状态

发生以下任何一种情况，设备退出 dStg 状态：

- 设备成功完成启动程序；
- 设备启动程序意外失败。

当设备退出 dStg 状态进入下一个状态之前，触发<dStgExitedEvent>事件。

一旦退出 dStg 状态，平台应记录一个信息，提示离开 dStg 状态。

10.2.8 dStp 状态

10.2.8.1 进入 dStp 状态

设备进入dStp状态有以下方法：

- 设备状态机的初始状态为 dStp;
- 设备正确关闭;
- 故障设备被排除。

设备进入dStp状态后触发<dStpEnteredEvent>事件。

10.2.8.2 dStp 状态过程

设备的启动、退出等程序因平台而异，因此本部分未定义。

10.2.8.3 退出 dStp 状态

当设备开始启动，退出dStp状态。

当设备退出dStp状态进入下一个状态之前，触发<dStpExitedEvent>事件。

10.2.9 dZom 状态

10.2.9.1 进入 dZom 状态

发生以下任何一种情况，设备进入dZom状态：

- 设备在其他任何设备状态下意外失败；
- 设备在最大允许时间内没有完成退出程序。

设备进入dZom状态后触发<dZomEnteredEvent>事件。

10.2.9.2 dZom 状态过程

一旦进入dZom状态，平台应执行以下操作：

- 记录一条信息，提示进入 dZom 状态；
- 清除所有与设备有关的平台资源；
- 释放设备的所有锁定；
- 设备当前的所有凭证全部失效。

一旦退出dZom状态，平台应记录一条信息，提示退出dZom状态。

10.2.9.3 退出 dZom 状态

当设备被回收，设备退出dZom状态返回dStp状态。当设备退出dZom状态进入下一个状态之前，触发<dZomExitedEvent>事件。

10.3 设备管理功能

10.3.1 设备安全性

设备不使用时，CUPPS平台应保证设备的逻辑性安全，具体要求如下：

- 设备分为已定义、必需、已定义但不必需三种。平台负责这些设备的操作控制，当设备不在正常使用状态时，平台应确保这些设备不能被操作。
- 对于未定义且不必需的其他设备，平台只确保工作站上的物理 I/O 端口安全。

10.3.2 图形化界面管理

平台应提供一个图形用户界面来管理设备。界面的功能是设备的配置和故障排除。

不对界面的功能作进一步说明。每个CUPPS提供商可从差异化竞争的角度自由实现任何功能。

10.3.3 日志

CUPPS平台应对设备相关操作实现全面的日志记录功能。记录包括传入、传出应用程序、平台设备组件的消息和数据流。

设备相关记录应遵循以下原则：

——平台可以打开设备相关记录；

——应用程序可以打开设备相关记录。

通过系统配置，设备相关记录可以被阻止，并且忽略其他平台和应用程序的开启要求。

11 应用程序需求

11.1 多核 CPU 需求

CUPPS应用程序应在多核CPU环境中正常运转。如果应用程序在多核环境中不能正确运行，应用程序提供商应通过Windows机制等方式解决这个问题。

11.2 CUPPSIT 信息工具

为了加强应用程序包的管理和可支持性，应用程序应提供输出信息的工具，该工具命名为 CUPPSIT。

CUPPSIT 工具应支持以下命令行参数：

——h 显示帮助信息；

——v 显示版本的可执行文件及其依赖性，输出应包括每个应用程序依赖性的版本。如果依赖信息输出，应使用的格式如下所示。输出应至少包括依赖项的名称和依赖项的路径；

——i 显示应用程序支持的接口标准；

——t[devicename] 发起指定设备的测试；

——d 显示应用程序适用的设备列表。

示例1：应用版本输出

```

1    C: > cuppsit -v
2
3    Application:  abc.exe
4    Version:     01.02.03
5
6    Dependency:  abc1.dll
7    Version:    01.02.03.1234
8
9    Dependency:  c: \path\to\abc2\abc2.dll
10   Version:    01.03.05.2029
11
12   Application:  abc2.exe
13   Version:     01.02.03

```

14

15 Dependency: abc1.dll

16 Version: 01.02.03.1234

17

18 Dependency: c: \path\to\abc2\abc3.dll

19 Version: 01.03.05.2030

20 C: >

注1: 示例中的可执行文件名和版本分别为abc.exe, v01.02.03和abc2.exe, v01.02.03。

注2: 应用程序决定输出的依赖项。

示例2: 应用帮助输出

1 C: > cupsit -h

2

3 Application: abc.exe

4 Version: 01.02.03

5

6 ABC's standard boarding Application. The following command line
7 parameters are supported

8

9 -t [printer] sends a test print to the specified printer

10

11 C: >

注1: 示例中输出的应用可执行文件的名称为abc.exe, 版本为v01.02.03。-t[device]参数由应用程序定义。

注2: 建议应用程序在帮助/关于菜单项中显示支持的接口标准。

示例3: 应用接口输出

1 C: > cupsit -i

2

3 Application: abc.exe

4 Version: 01.02.03

5

6 IATA-CUPPS versions: 01.00, 01.01, 02.00

7

8 C: >

注: 示例中输出的应用程序可执行文件名称为abc.exe, 版本为v01.02.03。应用程序支持国际航协CUPPS接口版本

01.00, 01.01和02.00。

示例4: 设备枚举功能

```

1    C: > cuppsit -d
2
3    Application:  abc.exe
4    Version:     01.02.03.0578
5
6    BC LHRT4LB00303BC1
7    PR LHRT4LB00303PR1
8
9    C: >

```

注: 示例列出了应用程序使用的所有设备的类型和名称。

示例5: 设备测试功能

```

1    C: > cuppsit -t LHRT4LB00302BC1
2
3    Application:  abc.exe
4    Version:     01.02.03
5
6    Testing:     BC LHRT4LB00302BC1
7    Result:      OK
8
9    C: >

```

注: 应用程序列出了设备测试的结果。结果显示的是平台API响应的res属性。

11.3 用户接口标准

应用程序的设计和功能由应用程序提供商和平台提供商控制, 但应符合以下要求:

- a) 应用程序应实现与底层操作系统厂商兼容的图形用户界面, 建议采用以下常用快捷键:
 - 1) 使用 F1 作为帮助键;
 - 2) 使用 ALT-F4 作为应用程序的退出键;
 - 3) 帮助/关于对话框提供下列信息: 版本信息, 支持的接口版本信息;
- b) 应用程序应实现基于英语的测试、界面管理或命令行。
- c) 工具应提供查看应用程序关键参数和测试设备信息等的基本功能。建议工具提供以下功能:
 - 1) 与上游主机和应用程序服务器系统的测试连接功能;
 - 2) 应用程序使用的设备的测试功能。
- d) 提供给管理员更多的工具, 以便更快速地诊断和解决问题。

11.4 日志

CUPPS 应用程序对应用程序的相关操作需要实现综合记录功能。记录包括传入、传出应用程序、其他主机相关通信的消息和数据流。

11.5 Windows 注册表约束性

由于Windows注册表的关键性，平台应适当控制对注册表的修改。注册表的可写区域有HKEY_CLASSES_ROOT、HKEY_CURRENT_USER、HKLM\Software\Classes、HKLM\Software\[Company Name] 和 HKLM\SYSTEM\CurrentControlSet\Services\[Service Name]。除非特殊情况，应用程序不能在注册表的任何地方写入。

11.6 应用程序包

为了促进应用程序包的有效交换，应用程序提供商和提供者应把应用程序打包在一个压缩的存档文件(ZIP文件)，并提供相应的MD5校验。ZIP文件的内容依赖于应用程序。脚本和其他ZIP文件要素应使用标准的环境变量。

