# 摘要

# 课题背景

Docker®容器通常用于在多台机器上无缝部署基于CPU的应用程序。有了这个用例，容器就是硬件无关的和平台无关的。使用NVIDIA GPU显然不是这样，因为它使用专门的硬件，并且需要安装NVIDIA驱动程序。因此，Docker Engine本身并不支持具有容器的NVIDIA GPU。

为了解决这个问题，出现的一个早期解决方案是在容器内完全重新安装NVIDIA驱动程序，然后在启动容器时传递对应于NVIDIA GPU（例如/dev/nvidia0）的字符设备。但是，该解决方案是脆弱的：主机驱动程序的版本必须与容器中安装的驱动程序版本完全匹配。 Docker图像无法共享，必须在每台机器上本地构建，从而打破了Docker的主要优势之一。

为了使Docker图像在使用NVIDIA GPU的同时便携式，nvidia-docker使用的解决方案是使图像与NVIDIA驱动程序无关。启动目标机器上的容器时，将安装所需的字符设备和驱动程序文件。

容器将应用程序包装到隔离的虚拟环境中，以简化数据中心部署。 通过包括所有应用程序依赖关系，如二进制文件和库，应用程序容器可在任何数据中心环境中无缝运行。

Docker是领先的集装箱平台，现在可用于集成GPU加速应用程序。 这意味着您可以轻松地集成和隔离加速应用程序，无需任何修改，并将其部署在任何支持GPU的基础设施上。 管理和监控加速数据中心从未如此简单。

# 文献综述

# 研究目标

本课题将研究基于金融IC卡标准的市民卡多应用平台的系统设计与实现。系统将实现IC卡应用生命周期管理、卡片生命周期管理、卡片安全域管理、灵活的行业应用接入、动态应用加载、并可对接其他TSM平台，如人行TSM、银联TSM、移动TSM等，实现应用共享和移动支付。

本课题将研究高性能计算系统在虚拟化技术中的实现。

# 研究内容

# 主要研究方案、技术路线与可行性分析

## 研究方案

## 技术路线

## 可行性分析

# 预期研究成果和（或）创新点

# 工作基础和条件

# 研究进度安排

# 参考文献

摘要

问题是什么。我们做了什么。大概怎么做的。做的还不错。

Abstract

1. 绪论
   1. 课题的背景介绍
   2. 国内现状与国外研究进展
   3. 论文主要目标
   4. 本文组织结构
2. 系统相关技术研究
   1. 虚拟化技术概述
   2. 基于Docker的容器技术
   3. 高性能计算平台
   4. 系统开发环境
   5. 本章小结
3. 高性能计算模型研究与设计
   1. CUDA编程模型
   2. 合作组织(CG)
   3. MPS
   4. 动态平行度
   5. NVML
   6. GPU P2P
   7. GPUDirect
   8. 本章小结
4. 系统在容器内的设计与实现
   1. 硬件平台结构
   2. Docker架构平台
      1. Docker概况
      2. Docker卷
      3. Docker扩展应用
   3. 软件系统组织结构
5. 算法在高性能计算系统中的实现与测试
   1. Cholesky分解算法实现
   2. LU分解算法实现
   3. PageRank算法实现
   4. Linpack测试容器内系统性能
6. 结论与展望
   1. 总结
   2. 展望