

基于数据集的BESIII数据 传输系统

张晓梅 林韬

IHEP

OUTLINE

- 目的
- 需求
- DIRAC开发简介
- 传输系统的设计
- 总结

目的——为何需要这样的系统？

- 传输系统的初衷：
 - 我们工作的环境是网格。
 - 在网格中，存储资源是不可靠的。
 - 传输文件时，文件可能会传输失败。
 - 如果传输批量的文件，那么手工管理将是噩梦。
 - 因此，需要传输系统。
- 为何需要设计一个新的传输系统？
 - 通用的传输系统过于通用
 - 我们有BESIII实验特有的需求
 - 整合已有的传输协议

需求

- 基本的功能
 - 将在其他站点中产生的模拟作业传输至高能所
 - 传输DST文件到其他站点
 - 基于数据集批量传输
 - 用户可以选择传输的协议
 - 查看分析历史
- 更高级一点
 - 监控数据集中文件传输的状况
 - 重新传输失败的文件
 - 管理员可以通过传输时的状态检测和分析站点的状态

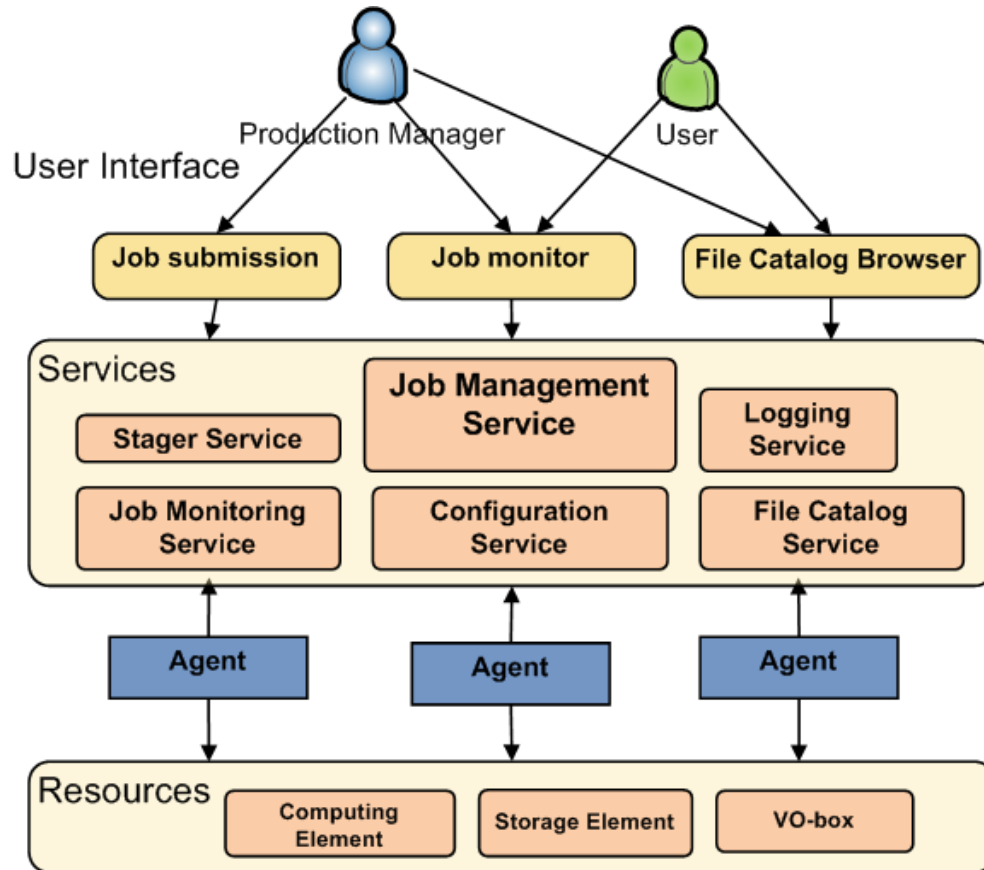
需求(cont.)

- 普通用户
 - 创建一个基于数据集的传输请求——create
 - 选择传输协议
 - 监测传输请求的状态——monitor
 - 对传输失败的文件进行重新传输——re-transfer
 - 删除误操作的请求——delete
- 管理员
 - 同意或拒绝传输请求——approve or reject
 - 检测站点之间传输的状态——health
 - 查看传输系统的历史信息——history
- 开发者
 - 复用已有的框架/模块
 - 安全，稳定
 - 具有扩展性
 - 与现有系统整合

基于DIRAC进行开发

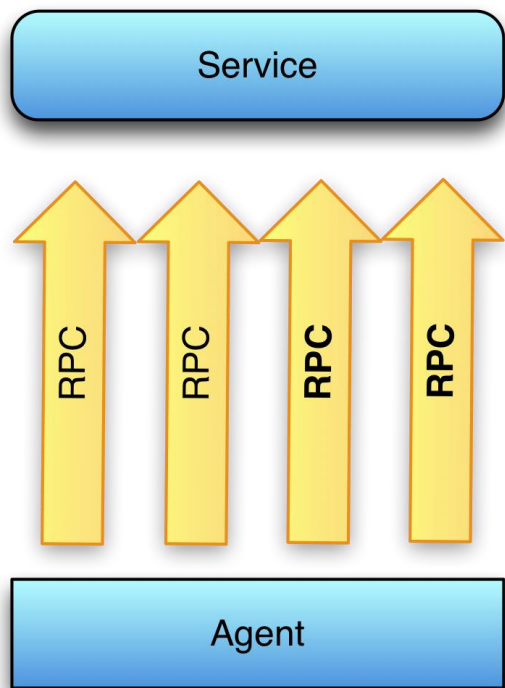
- 目前，BESIII实验采用DIRAC(Distributed Infrastructure with Remote Agent Control)作为网格的解决方案。
 - 该项目由LHCb发起。
 - 包含了**作业**和**数据**管理等系统。
- 遵循SOA(Service Oriented Architecture)范式
- 通过DISET实现远程通信。
- 在DIRAC系统中，包含：
 - **Service**（响应用户请求）
 - **Agent**（主动运行）
- 用户界面（User Interface）支持多种。
- DIRAC主要使用Python编写。扩展较为方便。
- 因此，传输系统**基于**DIRAC进行开发。

DIRAC基本架构

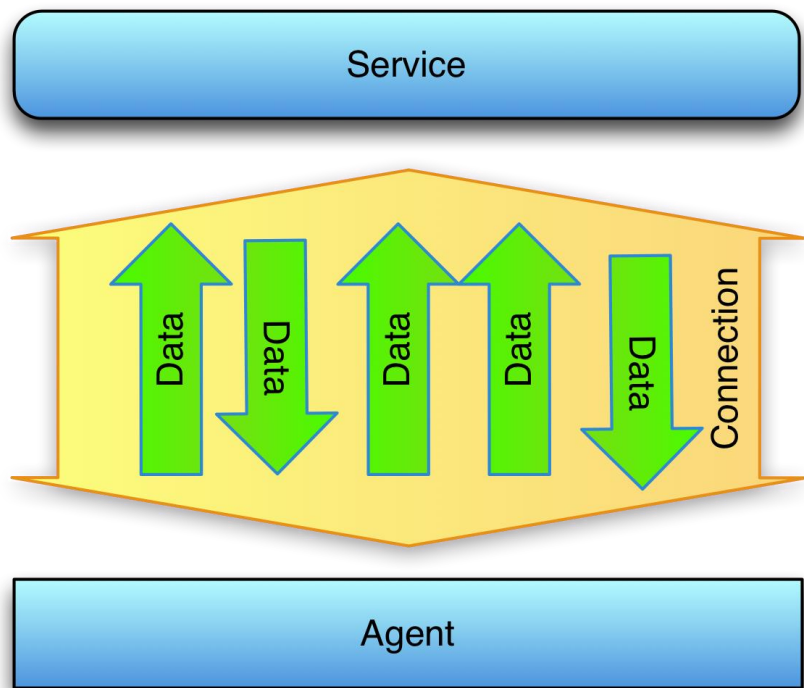


- Client与服务进行通讯——User Interface
- Service响应客户与Agent的请求
- Agent会自动执行特定功能

DIRAC服务端通信模式



同步模式

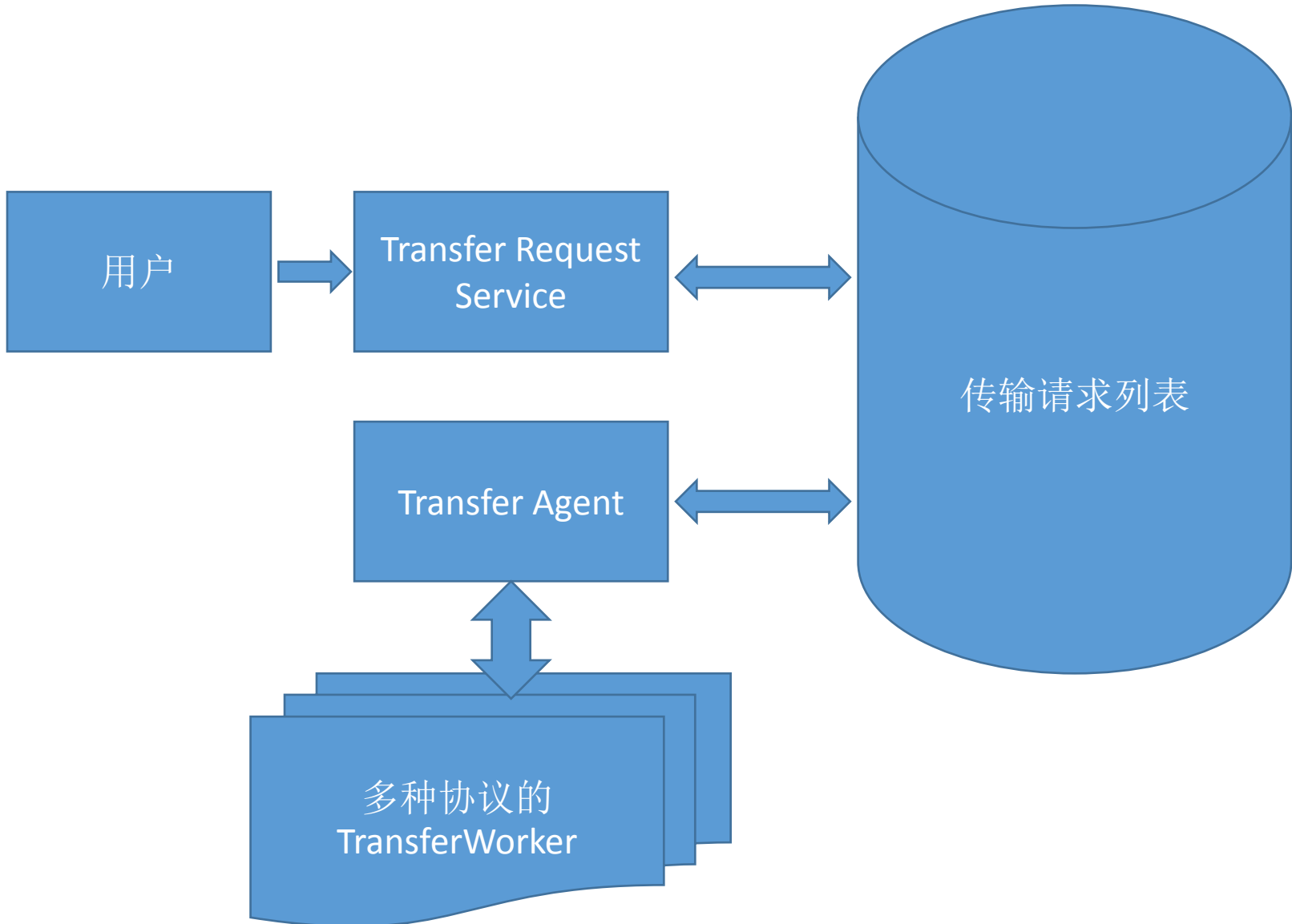


异步模式

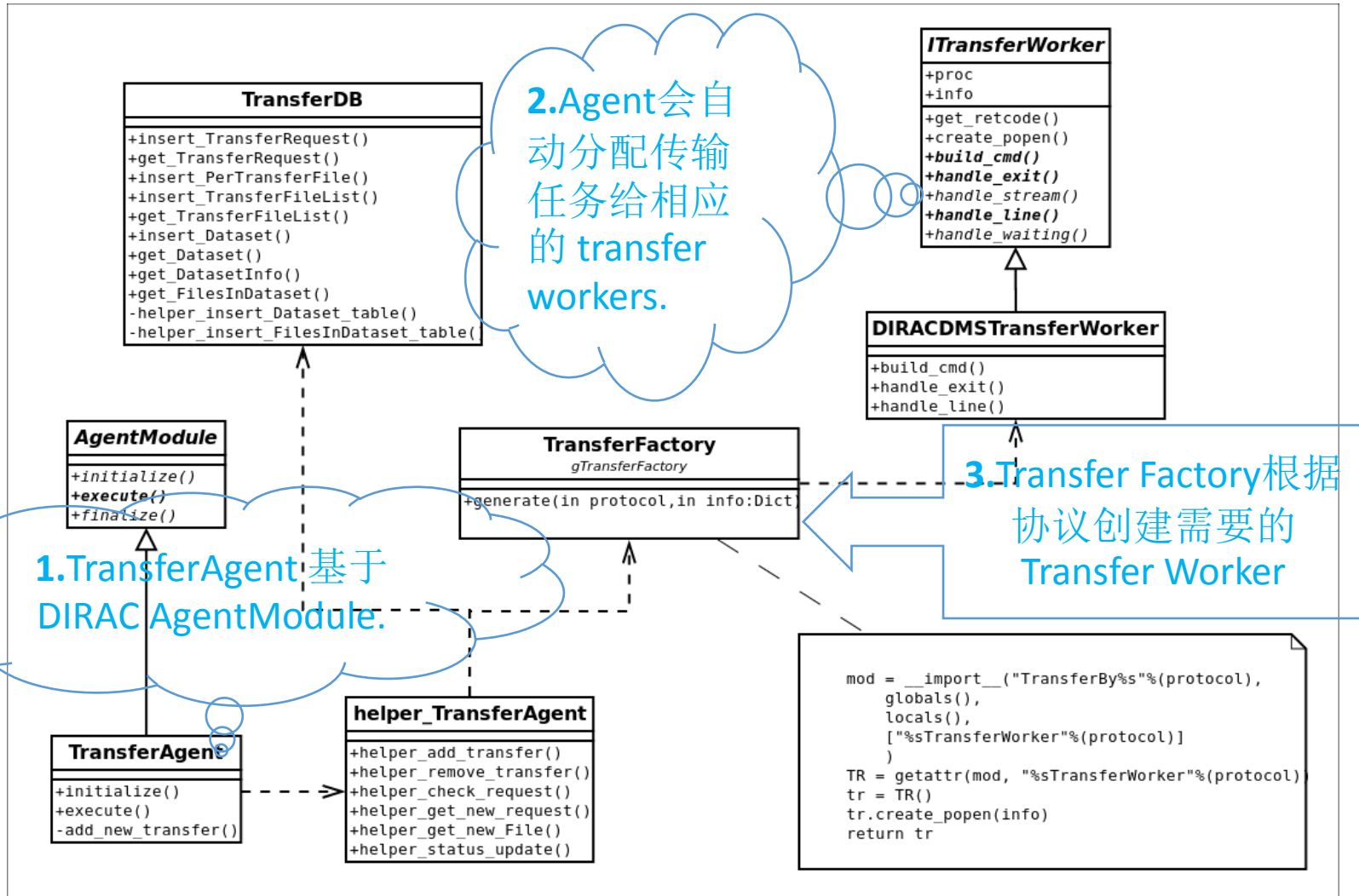
基于DIRAC的数据传输系统设计

- 传输系统包含：
 - Transfer Agent
 - Transfer Request Service
 - User Interface
 - Command line
 - Web
- Transfer Agent用于调度及运行传输请求。
- Transfer Request Service用于创建，修改及监控传输请求
- 两者之间通过数据库作为共享内存
- 引入一个工厂，用于创建不同协议的TransferWorker
- 使用异步I/O轮询子进程的状态

基本流程



Transfer Agent的设计



基于Web的UI（进行中）

- 服务端——Pylons
- 客户端——ExtJS
- 服务端用于转派客户端请求给DIRAC
- 客户端用于展示服务端返回的内容

System ▾ Jobs ▾ Views ▾ Bes ▾ Tools ▾ Selected setup: MyDIRAC-Production ▾

View Selected			Name ▾	Detail
	Function ▾	Script		
View	Status	bes-dirac-transf...		
View	ListRequest	bes-dirac-transf...		
View	ListFilelist	bes-dirac-transf...		

Page 1 of 1 | Displaying 1 - 3 of 3 | Page 1 of 1

DIRAC: v6r7p5 | lintao@ user ▾ (/C=CN/O=HEP/O=IHEP/OU=PHYS/CN=tao lin)

总结

- 传输系统的模型已基本建立
- 已实现SE之间的数据传输
- 不足：
 - 功能非常有限
 - 离投入正式使用有一定的距离
 - 用户界面要继续完善
- 因此，仍需继续开发，才能投入真正的使用之中
- 最终需要和BESDIRAC进行联调，保证BESDIRAC的稳定

Q&A