

WIPO | PCT

(84) 指定国(除另有指明, 要求每一种可提供的地区保护): ARIPO (BW, GH, GM, KE, LR, LS, MW, MZ, NA, RW, SD, SL, ST, SZ, TZ, UG, ZM, ZW), 欧亚 (AM,

图 2

- (57) Abstract:** Provided in the present invention are a cloud computing power allocation method, a user terminal, a cloud computing power platform and a system. Said method comprises: a user terminal generating a computing power request, the computing power request comprising a computing power demand and account information of a computing power scheduling center; sending the computing power request to a cloud computing power platform, so that the cloud computing power platform sends a configuration instruction to a computing device cluster according to the computing power request, the configuration instruction being used for allocating, to the user terminal, a target computing device which meets the computing power demand from the computing device cluster, and configuring, on the basis of the account information, the target computing device to be used for executing a computing task issued by the computing power scheduling center; and by using the account information, the user terminal acquiring computing power information from the computing power scheduling center, the computing power information being determined according to a computing result of the target computing device. By means of the solution, the complexity of a user acquiring the computing power is reduced, and the authenticity and reliability of computing power allocation is ensured.

AZ, BY, KG, KZ, RU, TJ, TM), 欧洲 (AL, AT, BE, BG, CH, CY, CZ, DE, DK, EE, ES, FI, FR, GB, GR, HR, HU, IE, IS, IT, LT, LU, LV, MC, MK, MT, NL, NO, PL, PT, RO, RS, SE, SI, SK, SM, TR), OAPI (BF, BJ, CF, CG, CI, CM, GA, GN, GQ, GW, KM, ML, MR, NE, SN, TD, TG)。

本国际公布：

— 包括国际检索报告 (条约第21条(3))。

(57) 摘要：本发明提供了一种云算力分配方法、用户终端、云算力平台和系统，其中该方法包括：由用户终端生成算力请求，算力请求包括算力需求和算力调度中心的账号信息；由向云算力平台发送算力请求，以使云算力平台根据算力请求向计算设备集群发送配置指令，配置指令用于在计算设备集群中为用户终端分配符合算力需求的目标计算设备，以及基于账号信息将目标计算设备配置为用于执行算力调度中心下发的计算任务；由用户终端利用账号信息从算力调度中心获取算力信息，算力信息根据目标计算设备的计算结果而确定。利用上述方案，降低了用户获取算力的复杂度，并且保证了算力分配的真实可靠性。