

12803031

ATA5DHB42/0001

保密信息

研发合作协议

项目编号:

合同编号:

研发合作协议

甲方：阿里巴巴（中国）有限公司

办公地址：杭州市西湖区西斗门路3号天堂软件园A幢10楼G座

联系人：肖司淼

工作邮件：simiao.xsm@alibaba-inc.com

乙方：南京大学

办公地址：江苏省南京市仙林大道163号南京大学仙林校区

联系人：黄宇

工作邮件：yuhuang@nju.edu.cn

鉴于，乙方在【数据库系统】领域有领先的技能、知识、专门知识和资源以及其他项目背景；

鉴于，甲方希望与乙方合作开展涉及【基于 RDMA 和 NVM 加速的 ParallelRaft 协议设计与验证技术研究】项目（“研发项目”）的工作；

据此，甲、乙双方经友好协商，就研发项目的相关事宜，达成如下技术研发协议（“本协议”），以兹共同遵守。

第一条. 定义

除非根据上下文应另作解释，本协议中使用的术语具有下文所载明的含义：

1. “知识产权”：指以下有形和无形的权益：与创作作品相关的权利，包括著作权、邻接权及其衍生作品的上述权利；商标、服务标记和商号权；商业秘密；发明专利、实用新型专利、外观设计专利及其申请以及其他工业财产权；域名；计算机软件，以及其他原因而产生的知识产权（如衍生产生的），无论是因法律运作、条约、合同、许可或其他方式，以及通过注册、初始申请、续展、延期、继续申请、分案申请或重授权前述任何一种知识产权而产生。知识产权包括但不限于专利申请权、专利权、使用权、许可权、转让权、收益权与诉讼权。

2. “商业秘密”：指一方的专有思想、可专利思想、现有或预期的产品和服务、软件、

电路图、研发、工艺、方法、流程、配方、公式、算法、参数、数据、结构设计、原理图、为合作研发设计的模具、生产、成本、利润信息、金融和金融项目、顾客、客户、市场等信息，以及当前或将来的商业计划和模型相关的所有非公开技术或商业信息，无论该信息是否与本协议相关。

3. “保密信息”：指一方向另一方提供、披露的，或一方所知悉或接触到的另一方之所有具机密或专有性质的商业和技术信息，包括但不限于本协议及其附件、本项目和补充协议的存在、条款及内容，以及因本项目所产生的所有知识产权、专有技术、可专利思想、现有或预期的产品和服务、软件、电路图、研发、生产、成本、利润信息、金融和金融项目、顾客、客户、市场等信息，结构设计、原理图、为合作研发设计的模具，当前或将来的商业计划和模型相关的所有非公开技术或商业信息，或任一方从第三方处获得但应承担保密责任的上述信息，皆统称为“保密信息”。不论该等信息是否被标示或专门指定为保密信息，也无论该信息是否与本协议相关，无论是以何种形式或载于何种载体，亦不论传达方式是书面或口头、磁盘、电子邮件等电磁纪录，以及是否为草稿或最终确认稿。

4. 开源软件：指需要（1）以源代码格式公布或分发；（2）或由用户授予第三方用来制作和/或分发衍生产品；或（3）免费重新分发才能使用、修改和/或分发的任何软件。此软件可以是由开源软件或其他软件整合而成、由开源软件派生而出或随同开源软件分发的软件。

5. “原有知识产权”：指由一方在履行本协议前所独立获取、付诸实施或持有的，用于本协议所开展的研发项目的知识产权。

6. “新产生知识产权”：指在本协议履行期间以及协议期满或终止后 90 天内，由甲、乙双方为了研发项目的结果而单独或合作研发完成的属于研发项目领域内的知识产权。“衍生知识产权”：在本协议履行期间以及协议期满或终止后 90 天内，由乙方单独完成的与研发项目相关，但不属于研发项目内也并非研发项目成果的知识产权。

7. “研发项目领域”：指涉及到本研发项目的研究、开发等所在工作领域，即【数据库系统】。

8. “研发项目”：指【基于 RDMA 和 NVM 加速的 ParallelRaft 协议设计与验证技术研究】。

9. “PI(Principal Investigator)”：指乙方为本研发项目所指定的项目负责人，在本协议中特指 黄宇，职务 教授，联系方式 yuhuang@n.ju.edu.cn。

10. “不可抗力”：指在本协议签署后发生的、本协议签署时不能预见的、其发生与后果是无法避免或克服的、妨碍任何一方全部或部分履约的所有事件。上述事件包括(a)自然灾害、战争、政府行为、国际或国内运输中断、流行病、罢工，以及根据中国法律或一般国际商业惯例认作不可抗力的其他事件；(b)因电力供应故障、通讯网络故障等公共服务因素或第三人因素；(c)在已尽善意管理的情况下，因常规或紧急的设备与系

统维护、设备与系统故障。

11. “关联公司”：指受一方直接或间接控制的实体、由该实体直接或间接控制该方，或者与该方受共同控制的任何其他实体。其中“控制”的含义是指直接或间接拥有该实体超过百分之五十（50%）有表决权的票证或该实体的其他所有权利益，或直接或间接拥有或控制该实体的权力，无论是通过拥有表决权的票证、协议或其他方式。

12. “控制权改变”：指一方的新设合并或被吸收合并入其他公司、单位或个人，或任何其他公司结构的重组，其中，在该新设合并、吸收合并或重组之后，分销商的资本股票代表该存续实体（或者，如该存续实体为全资子公司，其母公司）少于百分之五十（50%）表决权；或者，任何以分销商为一方的交易或一系列相关交易，其中，分销商超过百分之五十（50%）的投票表决权被转移到第三方；或者完成出售全部或绝大部分分销商资产的任何交易或一系列关联交易，除出售全部或绝大部分分销商资产给其他实体，分销商的股东拥有超过百分之五十（50%）具有表决权的合并表决权与其在该出售前所拥有的有表决权的票证比例大致相同。

13. “政府部门”：指中国或中国以外的任何国家级、州级、省级、地方或其它政府、政府性、管理性或行政性的政府部门、机构或委员会或任何法院、法庭或司法或仲裁机构。

14. “天”：指除本协议特别标明为工作日，指日历天。

15. “工作日”：指除星期六、星期日和其他法定节假日之外的任何一天，也包括原为星期六、星期日但已被国家假日办或其它有权机关调整为工作日的时间。

第二条. 研发人员及研发内容

1. 本协议生效后，乙方应在PI的安排下根据研发项目需要指派适合、足量的研发人员组成项目组具体负责项目研发，并向甲方提供项目组人员名单及相关信息；项目组经甲方确认后开展工作，研发过程中乙方如需调整项目组成员，需事先取得甲方的书面同意，且乙方应确保上述项目组成员均为乙方的在职人员及在校生，并应确保上述人员明确了解并同意受本协议的约束，尤其是知识产权、保密义务、文章发表的约定。

2. 双方应按照经双方确认的“研发项目说明书”（详见本协议附件二）的约定，按时、保质、按约推进研发项目的实施。研发项目说明书应详细描述本次研发项目的内容、双方的项目组成员、双方的研究力量和资源分配、实施计划、阶段性成果、验收标准等内容。

3. 双方经协商一致后同意调整研发范围的，应双方书面确定。甲方有权依据研发项目说明书监督、检查项目的执行，并根据监督、检查的结果要求乙方调整项目进度或撤换研发人员。

4. 乙方项目组主要负责利用乙方的学术资源和设施开展研发项目；甲方将根据研发项目需要，为乙方提供【经费、业务场景、必要的研发资源等】进行科学研究。

5. 甲方将视研发项目的必要性，派遣相关员工前往乙方工作场所与乙方项目组共同开展研发工作；期间，甲方员工的薪资、社保、差旅等费用均由甲方或甲方员工自行承担。

6. 本协议有效期内，甲乙双方应建立日常沟通机制，双方的项目负责人应至少每个月一次及时沟通项目实施情况，反馈需要解决的重点、难点和执行中的变化，并商讨下阶段的工作进展和结果。甲方项目组负责人如需变更，需事先书面告知乙方。

第三条. 研究经费及支付

1. 基于完成本协议下的研发任务，甲方将根据研发项目需要，为乙方提供研究经费（详见本协议附件一）及甲方同意的资源进行科学研究。

2. 甲方支付完毕所有研究费用后，即视为完成了本协议项下的全部支付义务。乙方与具体负责本研发项目的乙方项目组成员之间的费用分配、支付，以及由此产生的任何纠纷或争议，均由乙方自行处理，且不得影响本协议的执行，否则乙方应对由此造成的甲方的损失承担责任。乙方有义务向甲方提供符合甲方要求格式的费用明细汇总。

3. 乙方范围内的政府收缴税款由乙方缴纳。

4. 利用研究费用，用于乙方进行技术设计和实验而购置的设备、器材和资料归乙方所有。

第四条. 知识产权

1. 原有知识产权

双方在本协议履行之前各自拥有的原有知识产权仍归各自所有，如研究项目中不可避免必须要使用甲方的原有知识产权，经甲方书面同意，乙方仅能为执行研究项目在其内部于项目执行期间使用该原有知识产权。乙方同意就研究项目所涉及的乙方的原有知识产权，授予甲方及其关联公司在世界范围内不可撤销的、永久的、免费的制造、使用、修改、运行、演示、许诺销售、销售、制作衍生作品的权利以及对上述权利进行分许可的权利。

2. 新产生知识产权

除法律另有规定，新产生知识产权为甲方所有。甲方授予乙方在世界范围内不可撤销、永久的、免费的为科研、学术目的的内部使用许可。

双方根据研发项目的需要，建立定期例会机制，汇报研究项目的新产生知识产权和能够成为知识产权的创意、构思、技巧等，必要时经过双方商议，可临时组织交流会议。

3. 新产生知识产权的保护

新产生知识产权的获得、维护、保护等费用由甲方承担。双方应积极鼓励、协助和配合获得、维护和保护新产生知识产权，包括设置激励机制，以鼓励研究项目的参与人员积极参与提交专利申请、软件著作权登记等，以全面保护研究成果，还包括配合签署权利获得、维护的相关文件。

4 衍生知识产权及保护

在研究项目之外的衍生知识产权，甲方享有被授予该等衍生知识产权的许可权利，若乙方转让该等衍生知识产权，甲方享有优先受让权。就衍生知识产权的许可、转让的具体事宜，双方协商一致后，另行签署书面协议进行约定。

第五条. 文章发表

1. 经甲方事先审核与书面同意，乙方对就研究项目的成果可在国内或国际讨论会、会议、期刊、学位论文发表学术论文。发表时应标明甲方为研究项目的合作方。

2. 为了避免通过上述发表公开披露信息造成甲方信息的不恰当披露或知识产权保护的损失，乙方应在提交发表信息前至少 60 天，向甲方提供所要发表的所有材料（以下简称“发表材料”）。甲方在接到上述发表材料之后，应在 30 天内进行审核，逾期未审核视为同意。如甲方为申请合适的专利申请或采取其他方式来保护新产生知识产权或提出合理理由认为发表材料包含甲方保密信息，提出反对发表、删除保密信息或指出延后发表的合理意见，乙方应采纳甲方意见。

第六条. 保证和承诺

1. 双方保证与承诺其具有全部权利、能力和权限履行其于本协议下的义务。

2. 乙方保证与承诺将以专业、勤勉、适格、守时方式履行研发项目的研发工作及相关义务。

3. 乙方保证与承诺就本协议研发项目所产生的所有成果，或其它由履行本协议导致的成果均为乙方原创，均未侵犯任何第三方的知识产权及合法权益。

4. 就乙方向甲方许可的乙方原有知识产权和衍生知识产权，以及交付甲方所有的新产生知识产权，乙方保证与承诺其是该等知识产权的合法所有权人或享有分许可权利的被许可人，或可合法将该等知识产权交付甲方所有，且该等知识产权处于有效状态。该等知识产权上未设立包括质押发生在内的任何权利负担，当前没有实施该等知识产权侵犯第三方权利的未决侵权主张的情形，也没有关于该等知识产权有效性的未决争议。甲方根据本协议行使该等知识产权的行为不会违反其作为协议一方的任何协议下的义务。

5. 乙方保证与承诺，如甲方及其关联公司受到因行使本协议项下涉及的乙方原有知识产权、新产生知识产权、或衍生知识产权所导致的任何第三方对甲方及其关联公司的任何索赔、主张、诉讼，甲方有权立即终止本协议，乙方应该协助调查，如确属乙方责任，乙方应承担相关赔偿责任，赔偿总额不超过合同总额，乙方在本协议履行过程中存在故意或重大过失导致第三方索赔的除外。

第七条. 保密条款

1. 各方在讨论、订立及履行本协议中，对于一方（“披露方”）向另一方（“接收方”）提供的保密信息，未经披露方事先书面授权，接收方不得以任何方式向任何其他第三方泄露、转让、许可使用、交换或与任何第三方共同使用或不正当使用。

2. 关于本协议及其附件和补充协议的具体内容，除因履行应当履行的司法义务外，双方均同意未经对方事先书面许可，任何一方不得向第三方提供或披露。

3. 接收方不得对保密信息进行复制，逆向工程，反编译，试图破译源代码及潜在信息。

4. 保密义务期间双方有义务采取有效保密措施，接收方对因工作职责需要而知悉保密信息仅向该方的雇员进行披露。该等被披露保密信息的雇员均已签署内容实质与本协议本条规定相类似的保密协议，对保密信息承担保密义务。如根据法律要求、证券交易所的规则或法院要求接收方披露保密信息，接收方应给予披露方合理的事先通知并申请适当保护措施。

5. 如根据法律、法规或规章所需进行披露，接收方应使用所有适合的可用法律途径避免此披露，且在法律、法规或规章要求此披露前，尽最快可能通知披露方，由此披露方可寻求合适的保护途径或其他救济。

6. 接收方未经授权使用披露方的保密信息或对外披露都将给披露方造成不能挽回的损失和重大的侵害。上述未经授权的使用、披露将使披露方或构成不正当竞争的第三方对披露方造成的损害都将是不可挽回的损害。除所有法定的赔偿外，披露方将有权基于合理的判断而对任何实际或可能发生的违反本协议的行为，向有管辖权人民法院申请禁止令或适当的救济。就违约行为或强制性义务或未经授权使用或披露秘密信息行为而对披露方可能受到的全部损失、侵害（包括但不限于律师费用），接收方应承担足额的赔偿责任。

7. 本保密条款的保密义务不因本协议的变更、解除、终止而终止，该保密义务于披露之日起至被披露方公开之日止。

第八条. 对外宣传

1. 本协议履行期内，未经相对方预先审核和书面批准，任一方不得在经营活动中使用、

展示相对方及其关联公司的商标、LOGO、标识或通过特定描述使第三方能够识别相对方或其任何关联公司。

2. 未经过相对方事先书面同意，本协议任一方不得就双方针对本协议的合作关系、未来可能或实际进行的合作及项目进行对外公开、宣传和发表公开言论。如果就特定事宜需要进行对外公开、宣传和发表公开言论，双方须事先书面同意对外公开、宣传和发表公开言论的内容及渠道。

第九条. 违约责任

1. 如果任何一方未按照本协议的约定履行其义务，即构成违约。违约方应按照守约方的实际损失承担全部违约责任，且该违约行为在守约方发出改正通知之日起 30 天内未能纠正并获得守约方认可的，守约方可以书面通知方式终止本协议。

2. 如乙方未能按照本协议及其附件的约定完成研发项目，或有其他违约行为时，甲方有权不予支付相应阶段的研发费用，并有权视违约情形要求乙方承担相应的违约责任；如导致甲方将依第九条第 1 款终止本协议的，乙方将向甲方偿还至乙方接收到甲方发送的终止本协议的通知日期时未使用的研发费用，且已使用的部分应提供相应的凭证。

3. 违反本协议第三条约定，甲方应承担违约责任，每逾期支付一天，按逾期金额的 2% 支付逾期违约金，乙方计划顺延，逾期 30 天不支付经费，甲方除补交经费及逾期违约金外，同时支付协议额 10% 的违约金，乙方有权解除协议。

4. 乙方的 PI 或其他研究人员如存在任何违约行为，乙方应承担连带责任。

第十条. 通知和送达

1. 甲乙双方各指定本协议首页联系人为双方联系人，除本协议另有约定的除外本协议项下的通知、文件、资料等，均以信件邮递（包括快递、挂号信等）、电子邮件等书面形式发往对方。经对方书面确认，一方可对指定联系人和联系方式进行变更。

2. 本协议项下的通知或其他信息以电子邮件、传真发送的，发送方应及时提醒对方查收，得到对方书面确认则视为接收方收到通知；以快递或挂号信发送的通知，接收方签收则视为已收到通知。

3. 非向对方指定的联系地址、电子邮件发出的任何通知，视为自始未发出；任何一方应对其指定的联系人职责范围内的行为负责。

第十一条. 不可抗力

1. 如果发生不可抗力事件，为获得该免责，遭受不可抗力影响的一方应 1) 立即采取措施避免损失的扩大和因本协议不能或延迟履行而给对方造成的损失；2) 立即以可

能的最为快捷的方式通知对方并告知预计的恢复时间；3) 在不可抗力事件发生 15 天内向对方出具经当地公证机关公证、能有效证明该不可抗力事件发生的文件。不可抗力因素致使一方无法继续履行本协议或出现延误，完成上述义务后，该方不承担无法继续履行本协议或出现延误的违约责任。

2. 如不可抗力事故连续 60 天以上时，任何一方有权以通知方式终止本协议。

第十二条. 协议期限和终止

1. 本协议自双方加盖公章或合同章之日起生效，有效期【1 年】。

2. 本协议在下列情况下提前终止，相关方承担相应责任（如有）：

(1) 双方书面确认本协议提前终止时；

(2) 双方履行完本协议下各自的权利义务；

(3) 如本协议任意一方提出书面终止本协议意向，另一方在五（5）日内无书面应答或双方在三十（30）日内无法就继续履行本协议达成一致时；

(4) 双方中任何一方发生歇业、被吊销营业执照或进入破产、清算程序的，另一可书面通知解除本协议；

(5) 如不可抗力事件连续 60 天以上时，任何一方有权以书面通知方式终止本协议；

(6) 一方违反本协议的约定，该违约行为在守约方发出改正通知之日起 30 日内未能纠正并获得守约方认可的，守约方有权以书面通知方式终止本协议；

(7) 双方中任何一方发生控制权改变；

(8) 乙方的 PI 或重要人员发生重大变化，导致研究目的无法实现，经过合理努力，30 日双方不能达成一致继续履行本协议，甲方有权以书面通知的方式终止本协议；

(9) 乙方违反知识产权不侵权担保的约定，甲方有权以书面形式通知的方式终止本协议。

3. 本协议终止后，本协议保密条款、知识产权条款、保证和承诺条款、保密条款、对外宣传条款、通知和送达条款、文章发表条款、法律适用与争议解决条款以及和其他从属性上不应受本协议终止影响的条款继续有效。

第十三条. 法律适用与争议解决

1. 本协议的有效性、解释、执行及争议解决均适用中华人民共和国的法律和法规。因履行本协议产生的或与本协议有关的任何争议，应由甲乙双方友好协商解决；协商不成的，任何一方有权将争议提交原告所在地人民法院诉讼解决。

第十四条. 其他条款

1. 除本协议明确约定，任何一方并未明示或暗示赋予另一方的知识产权许可或其它任何权利。

2. 甲方有权以提前三十（30）日书面通知的方式将本协议的权利义务全部或者部分转让给甲方的关联公司，且该关联公司同意受本协议条款和条件的制约。未经过甲方事先书面同意，乙方不得以公开或隐蔽的方式，通过法律运作、合并（无论接收方是否是保留机构）或收购，或其它任何此类操作，转让本协议中的任何权利或职责。

3. 任一方不得直接或间接地向另一方及其关联公司之员工（包括但不限于全职及兼职员工、外包服务人员等）、顾问提供任何形式的不正当利益，不正当利益包括但不限于：（1）提供任何私人利益或赠与，包括但不限于实物、现金或现金等价物、优惠，以及其它财产性权益；（2）提供娱乐及款待；（3）提供工作机会；（4）提供投资机会；（5）提供借款。如一方知悉或怀疑另一方员工有违反前述约定的，有权与涉嫌违约方的廉政负责人或合规负责人联系，涉嫌违约方承诺对此种联系及相关信息进行保密。一方违反本条前述约定的，则任一其他方有权立即解除本协议，并由违约方按(a)本协议下对应的本项目甲方已支付的总价款的百分之三十（30%），或(b)提供任何形式的不正当利益的总金额，两者中较高的一项向守约方赔偿。

4. 双方应遵守所有适用的进出口法律法规、命令和政策（包括但不限于中国、美国及相关国际组织的进出口管制法律和条约）的相关规定，在必要的或对方请求的情况下，双方应相互提供给对方必要的合规协助。

5. 本协议附件是本协议不可分割的组成部分，与本协议具有同等法律效力。若附件和本协议的任何条款有冲突，以本协议条款约定为准。若具有合法管辖权的法院判定本协议任何规定无效，该判定不应影响本协议剩余规定，且本协议剩余规定仍应继续具有充分法律效力。若失效或无法执行条款系与定制产品直接相关，或与知识产权与保密等规定相关，甲方可自行决定是否要立即终止本协议。

6. 任何一方在任何时候放弃本协议项下的一种权利或放弃追究对方违约责任的权利，不被视为放弃本协议下其他权利或放弃追究对方其他违约责任的权利。

7. 本协议附件与本协议构成完整协议。本协议未尽事宜，可另行订立补充协议。双方确认，对本协议内容做出的任何修改和补充应为书面形式，由双方盖章后成为本协议不可分割的部分。

8. 本协议及其附件为双方基于本研发项目所签署的全部协议，应取代之前双方以任何行使达成的口头的、书面的，明示的、暗含的承诺、协议和保证等。

9. 本协议中的标题仅为方便而设，不对本协议的条款有任何限制作用，亦不具有任何法律效力。

10. 甲乙双方均为独立的缔约方且任一方无权代表另一方设置任何义务。本协议并不产生合资、合伙、雇佣或代理关系。

11. 本协议一式四份，双方各持两份，具有同等的法律效力。

附件一 支付

附件二 项目说明书

【以下无正文】

甲方：阿里巴巴（中国）有限公司

项目负责人（联系人）：陈森

盖章：

日期：2021.1.29



乙方：南京大学

项目负责人（联系人）：黄宇

盖章：

日期：2021年1月12日



Handwritten signatures and initials in black ink.

附件一 支付

1. 甲方将按照以下计划分【3】次向乙方支付研发费用，共计人民币（含税）49.042万元（大写：人民币 肆拾玖万肆佰贰拾元整）（以下称为“研发费用”）：

第一次：本协议生效、研究项目正式启动后且收到乙方开具等额正式增值税专用发票后30日内，甲方支付人民币（含税）19.042万元（大写：人民币 壹拾玖万肆佰贰拾元整）；

第二次：按照附件一约定，乙方完成 阶段1 和阶段2 的 阶段性成果，并向甲方提交 阶段1 和阶段2 的提交物，经甲方验收合格且收到乙方开具的等额正式增值税专用发票后30日内，甲方支付人民币（含税）15万元（大写：人民币 壹拾伍万元整）；

第三次：研发项目结项，乙方向甲方提交项目成果及结项报告，经甲方验收合格且收到乙方开具的等额正式增值税专用发票后30日内，甲方支付人民币（含税）15万元（大写：人民币 壹拾伍万元整）。

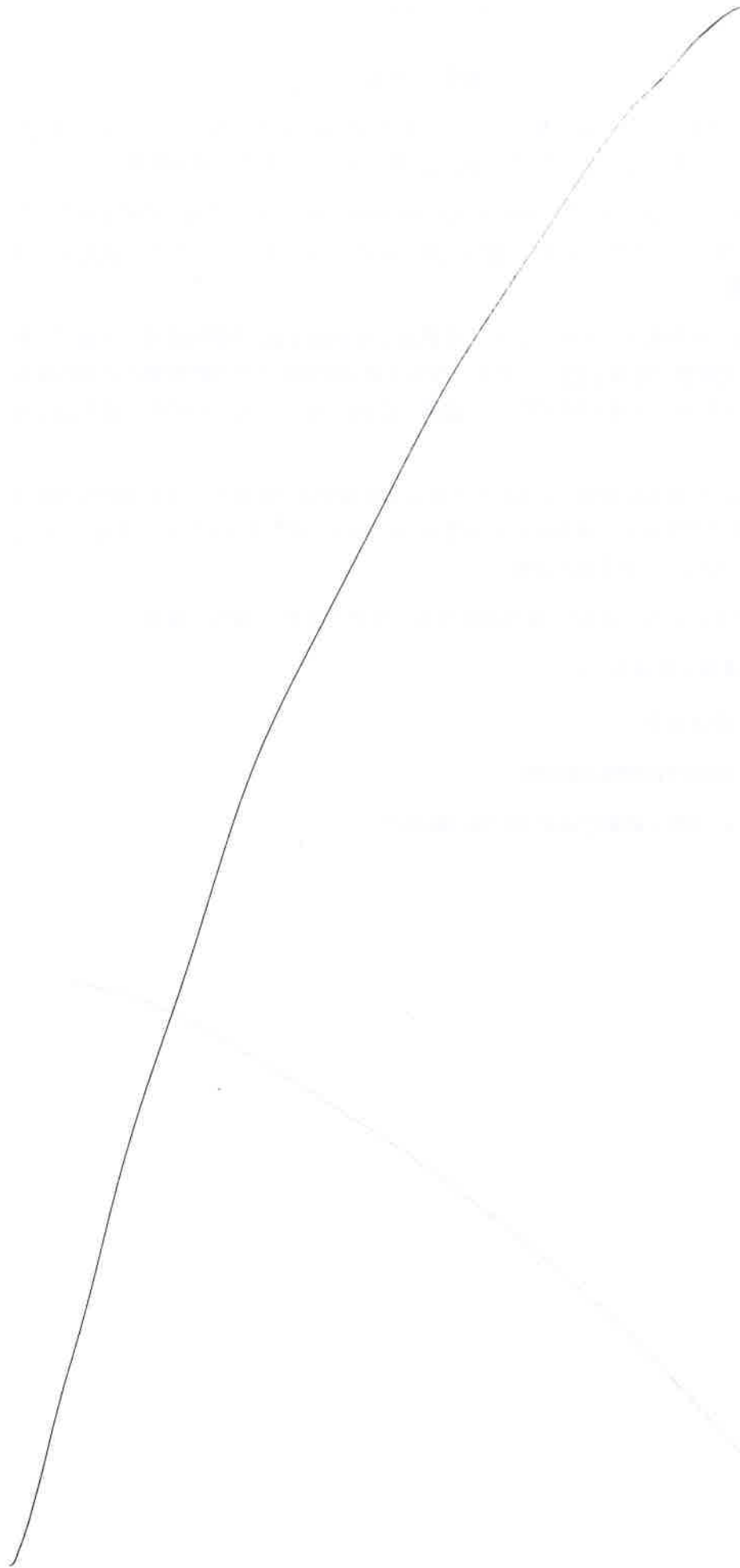
甲方应在乙方提交成果后30天组织验收，逾期不验收，视为已验收。

2. 乙方收款账户信息如下：

户名：南京大学

账号：4301011309001041656

开户行：中国工商银行南京分行汉口路支行



附件二 项目说明书

基于 RDMA 和 NVM 加速的 ParallelRaft 协议设计与验证技术研究

申请者姓名	黄宇
所属机构	南京大学
提交日期	2020.12.31

目录

1. 项目介绍

- 1.1. 题目
- 1.2. 项目背景及研究意义
- 1.3. 研究目标
- 1.4. 研究方法
- 1.5. 参考文献

2. 研究计划

- 2.1. 项目期限
- 2.2. 进度安排

3. 预期成果

4. 人力、设备等投入及项目预算

- 4.1. 人力投入
- 4.2. 设备投入
- 4.3. 项目预算

5. PI个人简历

1. 项目介绍

1.1. 题目

基于 RDMA 和 NVM 加速的 ParallelRaft 协议设计与验证技术研究

1.2. 项目背景及研究意义

PolarDB是阿里云自研的OLTP数据库,采用下一代的云原生架构,通过计算和存储的分离、物理复制技术等特性为客户提供了超高性能、超大实例、秒级快照、分钟级别扩缩容、按量付费等客户迫切需要的功能特性[1]。PolarFS是一个超低延迟、高吞吐量、高可用的分布式文件系统,它为上层的PolarDB提供共享存储设施[2]。由于端点失效在PolarDB集群中很常见,因此需要一个共识协议来保证所有被提交的数据修改(即使在各种边界情况下)都不会丢失。集群中的各个节点应该总能取得共识并达到数据的完全相同(bitwise identical),为此PolarFS使用经典共识协议Raft的变体ParallelRaft。ParallelRaft协议工作在PolarDB的存储节点(StorageNode)上,负责保护PolarDB的数据一致性、可靠性以及实例数据在存储节点群中进行一致性迁移调度的能力。ParallelRaft协议允许日志的乱序提交(commit),更重要的是它利用操作天然的可交换性允许日志中动作的乱序执行(execution),极大地提高了I/O操作的并发程度,进而提高了系统的吞吐率。

网络通信和数据存储设施是ParallelRaft协议运作的关键性硬件支撑,而这两类硬件设施的制造技术在不断发展,因而与底层硬件制造技术协同演进——充分发挥新硬件的性能优势,不断提升系统性能——是ParallelRaft协议乃至整个PolarDB系统的重要能力。具体而言,近年来RDMA(Remote Direct Memory Access)技术和非易失存储NVM(No Volatile Memory)设备的发展为ParallelRaft协议的设计带来了机遇和挑战。RDMA是一种新型网络技术,它可以绕过远端CPU直接读取远端内存,同时还具有高吞吐量和低时延的特性。这些特性使得RDMA技术可以有效地降低集群中数据同步的开销,因而RDMA技术在数据中心已逐渐普及。此处讨论的NVM特指新型非易失性存储设备,如AEP[3]、PCM[4]等。NVM能够在设备断电之后仍然保存其中存储的数据,同时具有可字节寻址、低时延、高密度等特点,其能耗比磁盘低90%以上。NVM在很多方面和DRAM(Dynamic Random-access Memory, 动态随机存取存储器)相似,尤其是其读写速度快、容量大、可字节寻址的特性,使其有望代替主存。NVM还具有DRAM所不具有的非易失性的特点,这使其可以在分布共识系统中用作日志等关键数据的持久存储,以提高共识系统的容错性。RDMA和NVM的结合可以提供更低延迟的可持久化硬件模型。ParallelRaft协议亟需紧随这波硬件技术的演进,构建下一代PolarDB数据库的存储节点,提供更低的I/O延迟,保持PolarDB在软硬件结合上的技术领先性。下面从新硬件带来的机遇出发,从共识协议的设计和正确性保障这两方面来分析现有ParallelRaft协议所面临的挑战。

1.2.1 RDMA和NVM的结合对ParallelRaft协议设计的挑战

从软件方法学的角度纵观分布共识理论与技术的发展历史,其基本设计原理自Paxos 算法提出与应用之后,并无重大变化[5]。其演化与发展更多是受到计算平台发展——包括计算设备、存储设备、网络通信设备等——与应用模式创新的推动,针对新计算场景做“派生式”(derivative)的演进[6]。网构软件方法学与技术体系着重关注软件的适应、演化与成长[7],以计算场景为核心抽象,从计算资源、计算环境、应用需求三个主要维度探讨软件技术的发展与基本软件范型的演变[8]。申请人所在的南京大学计算机软件研究所

吕建院士为学术带头人，是网构软件方法学与技术体系的主要提出单位之一。网构软件技术体系为新协议的设计提供了软件方法学层面的支撑。具体而言，我们从硬件设备的更新换代、计算模型的适应调整、协议设计的变与不变这三个层面来分析ParallelRaft协议设计所面临的挑战，如图1所示。

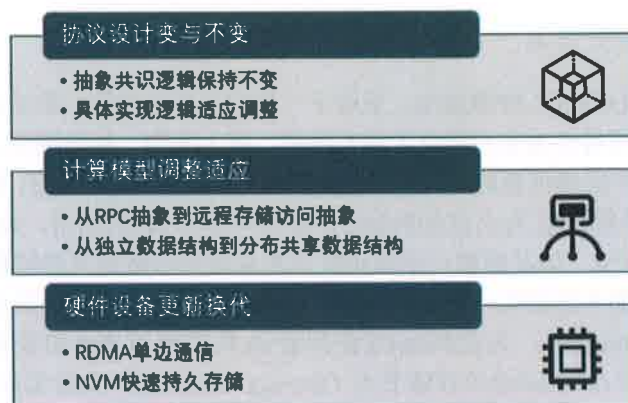


图1. 协议设计所面临的挑战

在硬件设备层面我们集中关注网络通信设备和数据的持续存储设备，它们分别对应于共识协议的两个核心需求，即数据的同步传输和持久存储。RDMA和NVM结合的新机遇在于，不仅可以绕过远端CPU实现易失数据的单边访问，而且可以实现NVM中持久数据的单边访问。基于这一特性，共识协议中领导者（leader）和跟随者（follower）的数据同步，特别是需要持久化的数据的同步，有望得到进一步的性能提升。此外，即使对于传统的需要远端CPU参与的协同逻辑——例如领导者失效的发现，新领导者的选举等——也有望利用NVM单边访问的性能优势获得性能提升。

从计算模型的角度看，RDMA技术不简单是数据传输速度的提升，它所支持的不唤醒对方CPU的单边通信方式，需要更准确的建模。传统网络环境中，领导者和跟随者之间的数据同步主要通过RPC的方式进行，其对应的计算模型抽象是消息传递模型。而RDMA所支持的单边通信，更适合使用共享存储模型来建模。更重要的是，目前主流的RDMA网络不支持持久存储的单边访问。NVM技术的更新换代使得持久存储的RDMA单边高速访问成为可能，这对系统的建模和后续的协议设计都是重要的机遇与挑战。在共享存储模型之上，构建分布式协议的核心问题是从简单到复杂、从底层到高层地设计不同的分布共享对象，有效支持上层共识协议的设计。

在协议设计层面，抽象的分布共识逻辑并无本质变化，而具体实现逻辑受到硬件升级变化的驱动，要做派生式的改进与优化。硬件设施、计算模型和协议设计的变化还需要我们对系统性能进行更全面的建模与更精细的分析。同时我们还需要原型系统的实现与实验，对协议改进及其系统实现进行更全面的评估。

1.2.2 ParallelRaft的优化对协议正确性保障的挑战

引入新硬件提升协议性能的同时，也带来了协议正确性保障的挑战。形式化方法可以对分布式协议的执行进行严格的数学建模，并通过模型检验等手段穷尽系统所有可能的执行，以确保协议的设计是正确的[9][10][11]。但是形式化方法的应用面临三个层次的挑战，如图2所示。

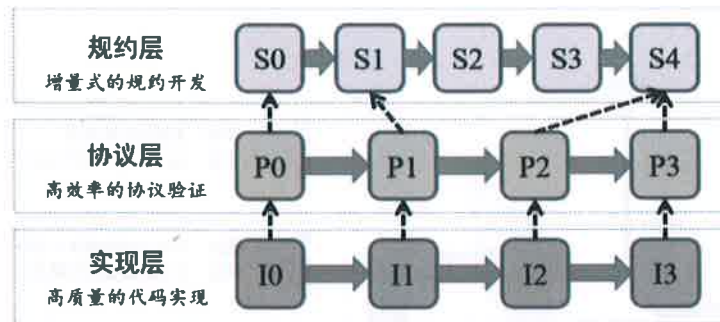


图2. 协议验证所面临的挑战

第一，在形式化验证过程中，形式化规约的开发过程与传统的软件开发类似，需要考虑规约开发的代价、规约的可维护性等传统软件工程中的基本问题。特别是对于协议的派生式设计，形式化规约的开发也需要是增量式的，能充分复用既有的规约，并能与协议的开发过程有机互动[12]。

第二，模型检验技术的应用存在固有的状态爆炸问题[13]。枚举系统所有可能执行的技术手段，往往仅能用于规模非常有限的系统设计中。对于一个真实的分布式协议而言，其模型检验的代价往往是不可接受的，必须针对性地提出应对、规避状态爆炸问题的技术手段。

第三，模型检验技术实际效用的一个重要局限在于它的正确性结论仅局限于其模型。而我们最终的目标是保障真实代码实现的正确性。抽象模型与具体实现之间存在较大的鸿沟，这可能严重削弱模型检验结果的实际效用[14]。

针对上述挑战，我们要系统地研究派生式共识协议优化的正确性保障问题。

1.3. 研究目标

本项申请的研究目标是：

- 在协议设计层面：针对RDMA结合NVM的硬件技术升级，派生式地改进和优化ParallelRaft协议的设计，将硬件的升级换代有效转化为系统性能的提升。
- 在协议正确性保障层面：针对协议派生式设计与演化的特征，增量式地进行协议的形式化规约，对协议的设计进行高效的模型检验，并基于模型检验的指导对协议的实现进行充分的测试，基于软件形式化方法有效保障新协议及其实现的正确性。
- 在软件方法学层面：深入理解分布共识协议演化成长内在的机理，有效指导场景驱动的派生式协议设计，力争在每次硬件设施、应用模式的升级换代机遇来临之际，总能领先竞争者一步，更快更好地实现共识协议及其系统实现的升级换代。

1.4 研究方法

本研究关注RDMA结合NVM新硬件驱动下的ParallelRaft协议设计，以网构软件方法学为指导，提炼共性不变的抽象共识逻辑，派生式地改进网络通信和持久存储相关的具体实现逻辑。进而引入形式化规约与验证技术，力求充分保障新协议的设计和实现的正确性。下面分别从新协议的设计和验证两个方面来讨论本项申请的技术路线和研究内容。

1.4.1 新硬件驱动的ParallelRaft协议设计

新硬件组合RDMA+NVM的出现，为上层协议的设计提供了更高效的底层通信和存储手段。协议的设计者因而需要改变计算模型的抽象以求更准确建模系统的运作机理，进而对上层协议进行派生式地设计改进。我们从硬件、模型、协议这三个层面分别进行阐述，

如图3所示。

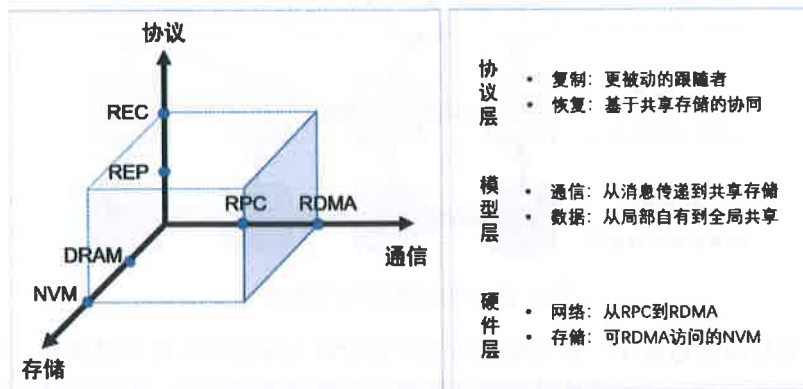


图3. 新硬件支撑下的共识协议设计

RDMA技术近年来已经在数据中心计算场景中逐渐普及。近期的一个重要的技术进展是非易失存储不仅速度逐步提升，而且也开始可以基于RDMA技术，不唤醒远端CPU地进行单边访问。RDMA技术的出现使得物理上分布的内存和持久存储，逻辑上更像是所有节点共有的，只不过不同节点访问不同存储的速度有差别，并且访问不同存储区域的权限可以有（精心设计的）差异。RDMA和非易失存储技术的上述进展是本项申请研究工作的核心推动力。

RDMA和非易失存储的上述进展，使得共享存储模型成为建模系统运作更准确的模型。在共享存储模型中，模型提供最基础的共享数据对象，协议的设计者通过设计抽象层次逐渐升高的不同共享数据对象，最终解决上层的分布式算法问题。常见的最基础共享数据对象是共享寄存器（shared register）[15]。每个共享寄存器具有不同的访问模式，主要包括单写多读（Single Writer Multiple Reader, SWMR）和多写多读（Multiple Writer Multiple Reader, MWMR）两种。每个寄存器在并发读写下遵循一定的一致性模型，例如linearizability等[15][16]。上层协议的运作是围绕多个共享存储上的不同抽象层次的共享数据对象进行的，例如共识协议本质就是对一个高层的共享日志（shared log）对象的读写。整个类Multi-Paxos协议族（包括Raft、ParallelRaft等）可以看成是shared log结构的维护[17]。

底层计算模型的更新，会带来上层共识协议设计的“变”与“不变”。面对硬件技术的更新换代，核心的抽象共识逻辑没有本质变化。协议的主要变化在于针对网络、存储硬件的更新换代，做派生式地改进和优化。从协议设计的视角看底层硬件，协议主要需要两方面的支撑。首先，共识过程中的一些数据需要持久存储，以保证在节点失效并恢复之后能够获取失效前的数据，例如共识过程中的一些关键性的元数据以及日志数据等。其次，共识协议需要保证多个节点的数据一致，这必须要依赖底层的网络通信设施的支持。

ParallelRaft协议采用强领导者结构，其运作围绕领导者的推动进行。强领导者结构自然产生两个关键问题，一个问题是如何高效地将数据从领导者复制到跟随者（图3中简称REP，即replication），另一个问题是领导者失效时如何正确迅速地选出新领导者、修复日志并继续正常运作（图3中简称REC，即recovery）。对于领导者向跟随者的复制数据的问题，应充分利用RDMA和非易失存储硬件的新特性，将跟随者放在更被动的位上，高效地完成领导者单边推动的数据复制。对于领导者失效恢复的问题，多个候选者之间需要以共享存储的读写为基本协同原语，实现领导者健康状况监控、新领导者选取以及日志的修复。在面临各种系统和网络失效时，基于共享存储的协同还需要与传统基于RPC通信方式的协同有机地进行结合。

底层硬件设施和计算模型的变化同样带来协议性能建模与分析的变化。新的共享存储模型不仅需要精准建模不同存储介质访问代价的差异，还需要建模易失存储访问速度的不对称，以及易失存储读写寿命等传统协议分析并未建模的问题。上述协议设计的工作最终将通过原型系统实现、演示和性能评估，综合论证其效率与正确性。

1.4.2 ParallelRaft协议优化的正确性保障

协议的优化带来性能提升的同时，也对其正确性保障提出了严峻挑战。软件形式化方法是我们保证协议及其系统实现正确性的重要手段之一。针对1.2节所述软件形式化方法在实用中面临的挑战，我们将从规约、验证、测试这三个方面分别展开关键技术的研究，如图4所示。



图4.共识协议优化的验证

在协议的形式化规约方面，针对协议的设计是派生式、增量式演进的特点，我们首先需要对改进前的协议的关键性模块作完整的形式化规约。其次对于协议的派生式优化，我们将主要依赖形式化规约中的精化（refinement）关系，通过对既有的规约做精化，来实现增量式、高效率的规约。具体的规约通过TLA+形式化规约语言[18]实现。

在协议的验证方面，我们基于协议的形式化规约，采用模型检验的手段来发现协议中潜在的bug。对于模型检验所固有的状态爆炸问题，我们对整个执行空间进行采样。为此要解决两个关键问题。首先我们要对采样所得的执行路径进行度量，度量的目的是科学定量地优先考虑对协议的验证/测试目标更为有利的执行路径。其次对于所设计的采样执行策略，我们要分析它对整个执行空间的覆盖程度，进而在资源消耗与关键执行逻辑覆盖方面做出高效的权衡。执行空间的采样不仅在协议层实现了高效的验证，而且它是后续测试用例生成的必要基础。此外我们还可以对协议的形式化规约进行定理证明，所证明的结论是与系统执行的具体规模无关的。

协议层/设计层形式化验证技术的正确性保障只能局限于协议设计所基于的抽象模型，对于协议的代码实现，理论上可以将代码作为模型，直接进行代码层的模型检验[19]，但是这一手段往往由于其复杂性过高而实际上难以广泛应用。我们主要采用基于模型的测试（model-based testing）技术保障其正确性，并借助Jepsen[20]等主流测试工具进行实施。基于模型的测试是我们形式化验证技术在代码层的延伸。具体而言，

- 首先我们研究基于模型的测试用例生成，从模型层的执行中分析出协议潜在缺陷的关键特征，并用脚本自动生成测试用例，辅助以环境错误的自动注入等技术，对协议的代码实现进行更具针对性的测试。
- 其次我们研究执行轨迹的验证。其基本方式是采用某种通用的模型检验器（例如TLA+工具集中的TLC），将执行轨迹作为模型，针对系统的正确性规约进行验证。由于系统的执行轨迹是由系统真实代码实现的运行产生，所以它能对代码实现的正确性提供保障。此外轨迹验证的规模从“考虑所有可能的执行”降低为“仅考虑一次具体的执行”，并且我们可以在轨迹记录时控制记录的精细程度，所以轨迹验证技术有望缓解和规避模型检验的状态爆炸问题。

1.5 参考文献

- [1] Feifei Li. Cloud-Native Database Systems at Alibaba: Opportunities and Challenges. *PVLDB*, 12(12): 2263 - 2272, 2019.
- [2] Cao W, Liu Z, Wang P, Chen S, Zhu C, Zheng S, Wang Y, Ma G. PolarFS: an ultra-low latency and failure resilient distributed file system for shared storage cloud database. *Proceedings of the VLDB Endowment*. 2018 Aug 1;11(12):1849-62.
- [3] 英特尔傲腾持久内存, <https://www.intel.cn/content/www/cn/zh/architecture-and-technology/optane-dc-persistent-memory.html>
- [4] 冒伟, 刘景宁, 童薇, 冯丹, 李铮, 周文, 张双武, 基于相变存储器的存储技术研究综述, *计算机学报*, 38(5):944-960, 2015年5月。
- [5] Dahlia Malkhi (Ed.). 2019. *Concurrency: the Works of Leslie Lamport*. Association for Computing Machinery, New York, NY, USA. DOI:<https://doi.org/10.1145/3335772>.
- [6] Yanhua Mao, Flavio P. Junqueira, and Keith Marzullo. 2008. Mencius: building efficient replicated state machines for WANs. In *Proceedings of the 8th USENIX conference on Operating systems design and implementation (OSDI'08)*. USENIX Association, USA, 369–384.
- [7] Hong Mei and Jian Lu. *Internetware - A New Software Paradigm for Internet Computing*. Springer Singapore, Singapore, 2016.
- [8] 曹春, 马晓星, 面向人机物融合应用的场景计算机, *中国计算机学会通讯*, 16(4), 2020.4.
- [9] Chris Newcombe, Tim Rath, Fan Zhang, Bogdan Munteanu, Marc Brooker, and Michael Deardeuff. 2015. How Amazon web services uses formal methods. *Commun. ACM* 58, 4 (April 2015), 66–73. DOI:<https://doi.org/10.1145/2699417>.
- [10] Chris Hawblitzel, Jon Howell, Manos Kapritsos, Jacob R. Lorch, Bryan Parno, Michael L. Roberts, Srinath Setty, and Brian Zill. 2015. IronFleet: proving practical distributed systems correct. In *Proceedings of the 25th Symposium on Operating Systems Principles (SOSP '15)*. ACM, New York, NY, USA, 1–17. DOI: <https://doi.org/10.1145/2815400.2815428>.
- [11] Chris Hawblitzel, Jon Howell, Manos Kapritsos, Jacob R. Lorch, Bryan Parno, Michael L. Roberts, Srinath Setty, and Brian Zill. 2015. IronFleet: proving practical distributed systems correct. In *Proceedings of the 25th Symposium on Operating Systems Principles (SOSP '15)*. Association for Computing Machinery, New York, NY, USA, 1–17. DOI: <https://doi.org/10.1145/2815400.2815428>.
- [12] Kleuker S., Tjabben H. (1996) The incremental development of correct specifications for distributed systems. In: Gaudel MC., Woodcock J. (eds) *FME'96: Industrial Benefit and Advances in Formal Methods*. FME 1996. *Lecture Notes in Computer Science*, vol 1051. Springer, Berlin, Heidelberg. https://doi.org/10.1007/3-540-60973-3_103.
- [13] Clarke E.M., Klieber W., Nováček M., Zuliani P. (2012) Model Checking and the State Explosion Problem. In: Meyer B., Nordio M. (eds) *Tools for Practical Software Verification*. *LASER 2011. Lecture Notes in Computer Science*, vol 7682. Springer, Berlin, Heidelberg. https://doi.org/10.1007/978-3-642-35746-6_1.
- [14] A. Jesse Jiryu Davis, Max Hirschhorn, and Judah Schvimer. 2020. Extreme modelling in practice. *Proc. VLDB Endow.* 13, 9 (May 2020), 1346–1358. DOI: <https://doi.org/10.14778/3397230.3397233>.
- [15] Lamport, L. On interprocess communication, Part II: Algorithms. *Distrib Comput* 1, 86–101 (1986). <https://doi.org/10.1007/BF01786228>.

- [16] Maurice P. Herlihy and Jeannette M. Wing. 1990. Linearizability: a correctness condition for concurrent objects. ACM Trans. Program. Lang. Syst. 12, 3 (July 1990), 463–492. DOI:<https://doi.org/10.1145/78969.78972>.
- [17] Mahesh Balakrishnan, Jason Flinn, Chen Shen et. al., Virtual Consensus in Delos, in OSDI 2020: 14th USENIX Symposium on Operating Systems Design and Implementation, November 2020.
- [18] The TLA+ homepage, <https://lamport.azurewebsites.net/tla/tla.html>.
- [19] Madanlal Musuvathi, David Y. W. Park, Andy Chou, Dawson R. Engler, and David L. Dill. 2003. CMC: a pragmatic approach to model checking real code. SIGOPS Oper. Syst. Rev. 36, SI (Winter 2002), 75–88. DOI:<https://doi.org/10.1145/844128.844136>.
- [20] Jepsen – Distributed Systems Safety Research, <https://jepsen.io/>

2. 研究计划

2.1. 项目期限：以实际合同签字生效日期为准，合同期1年。

2.2. 进度安排：

阶段	研究内容	时间	提交结果
1	ParallelRaft 协议的形式化规约和验证	第 1 季度	规约开发说明文档；TLA+规约代码；形式化验证结果；
2	ParallelRaft - Passive Follower 协议的建模和设计	第 2 季度	协议设计文档；协议伪码实现；协议正确性的非形式化论证和性能建模分析；
3	ParallelRaft - Passive Follower 协议的增量式验证	第 3 季度	规约开发说明文档；TLA+规约代码；形式化验证结果；系统设计和验证的研究成果形成数据库/存储领域 CCF-A 类论文投稿；
4	ParallelRaft - Passive Follower 协议的实现、实验和演示	第 4 季度	系统设计文档；系统源码；实验设计文档；实验数据与实验报告；

3. 预期成果

本项目研究预计产生如下研究成果：

- 1) 交付 Parallel Raft – Passive Follower 模型可演示的 demo 以及性能测试报告，要求系统 16KB 端到端的写延迟从 120us 降低到 90us，同时确保系统的 IOPS、RPO、RTO 等其它指标不受损；交付完整的模型描述文档；
- 2) 对 Parallel Raft 协议的模型和设计做完整的 TLA+建模，交付完整的，可维护的 Parallel Raft 模型的 TLA+代码；对改进后的 Parallel Raft-Passive Follower 模型和设计做完整的 TLA+ 建模，交付完整的，可维护的 Parallel Raft – Passive Follower 模

- 型的 TLA+ 代码;
- 3) 交付完整的 Parallel Raft – Passive Follower 模型的数学证明过程文档;
- 4) 数据库/存储领域 CCF A 类会议论文一篇;

4. 人力、设备等投入及项目预算

4.1. 人力投入 (包括全部参与项目的人员)

人员姓名	所属机构与职务	人员工资 (元/月)	参与项目时间 (折合全时人月数)	人员工资总计 (元)	承担职责	来阿里实习 时间
有固定收入教师的人员费用						
黄宇	南京大学 教授	12,000	10	120,000	PI	
魏恒峰	南京大学 副研究员	10,000	6	60,000	项目主要成员	
研究生的劳务费用						
谷晓松	南京大学 博士生	4000	12	48,000	项目主要成员	2021.6~2021.12
唐瑞泽	南京大学 博士生	4000	12	48,000	项目主要成员	2021.6~2021.12
杨一江	南京大学 硕士生	3000	12	36,000	项目主要成员	
侍林天	南京大学 硕士生	3000	12	36,000	项目主要成员	
黄彬寓	南京大学 硕士生	3000	12	36,000	项目主要成员	

4.2. 设备投入

设备名称	数量	价格 (元)	总计 (元)	备注

4.3 项目预算

类别	总计 (元)	备注
① 人员支出	384,000	与4.1对应

② 研发支出		
出差/会议费用	50,000	1. 国内差旅单趟5000, 国际差旅单趟15000, 以实际开销为准; 2. 学生来阿里实习期间的津贴、补助、保险等全部费用包含在此项经费内。
设备费		与4.2对应
其它费用		由项目PI根据项目情况来填
② 直接费用	434,000	①+②
③ 间接费用	56,420	学校学院管理费10%, 专票税费3%
总计	490,420	③+④

5. PI个人简历

黄宇, 南京大学计算机软件新技术国家重点实验室, 教授/博士生导师

1) 基本情况

现为南京大学教授、博士生导师, 主要研究方向为分布式算法、分布式系统软件、软件形式化方法。主持与参与国家自然科学基金项目、国家重大基础研究计划(973)等多项国家级科研项目。在 ACM PODC、IEEE SRDS、IEEE TC、IEEE TPDS 等国际会议/期刊上发表学术论文多篇。

2014 年获南京大学登峰人才支持计划资助, 2018 年获首届高校计算机专业优秀教师奖励计划奖励, 指导的博士论文获得 2016、2017 年度中国计算机学会优秀博士论文奖(与吕建教授联合指导)。

2) 教育/研究经历

- 2016-12 至今, 南京大学计算机系, 教授
- 2010-12 至 2016-12, 南京大学计算机系, 副教授
- 2007-7 至 2010-12, 南京大学计算机系, 讲师
- 2005-9 至 2006-9, 香港理工大学电子计算学系, 研究助理
- 2002-9 至 2007-6, 中国科学技术大学计算机系, 硕博连读(导师: 冯玉琳)
- 2003-9 至 2007-6, 中科院软件所, 软件工程技术中心, 联合培养博士生学习
- 1997-9 至 2002-6, 中国科学技术大学少年班(计算机软件专业), 本科学习

3) 主要研究成果(近 5 年)

申请人主要研究分布式算法、分布式系统的形式化规约与验证理论和技术。近期主要关注分布共识协议和分布事务并发控制协议的规约与验证。主要基于 TLA+规约与验证工具集, 着力在真实的大规模分布式系统中推广应用形式化规约与验证技术。基于形式化验证技术在代码层应用的局限, 进一步研究基于模型的测试技术, 用协议层验证信息指导代码层测试, 有效提升了真实系统实现的可靠性。

近 5 年相关研究的论文与专利如下:

1. Kaile Huang, Yu Huang, Hengfeng Wei, Fine-grained Analysis on Fast Implementations of Distributed Multi-writer Atomic Registers. PODC 2020: 200-209.
2. Hengfeng Wei, Yu Huang, Jian Lu, Brief Announcement: Specification and Implementation of Replicated List: The Jupiter Protocol Revisited. PODC 2018: 81-83, 2017.
3. Hengfeng Wei, Yu Huang, Jian Lu, Probabilistically-Atomic 2-Atomicity: Enabling Almost Strong Consistency in Distributed Storage Systems. IEEE Trans. Computers 66(3): 502-514 (2017).
4. Yiling Yang, Yu Huang, Xiaoxing Ma, Jian Lu, Enabling Context-Awareness by Predicate Detection in Asynchronous Environments. IEEE Trans. Computers 65(2): 522-534 (2016).
5. Hengfeng Wei, Marzio De Biasi, Yu Huang, Jiannong Cao, Jian Lu, Verifying Pipelined-RAM Consistency over Read/Write Traces of Data Replicas. IEEE Trans. Parallel Distributed Syst. 27(5): 1511-1523 (2016).
6. Lingzhi Ouyang, Yu Huang, Hengfeng Wei, Jian Lu, Achieving Probabilistic Atomicity With Well-Bounded Staleness and Low Read Latency in Distributed Datastores. IEEE Trans. Parallel Distributed Syst. 32(4): 815-829 (2021), 2020.
7. Xue Jiang, Hengfeng Wei, Yu Huang, A Generic Specification Framework for Weakly Consistent Replicated Data Types. SRDS 2020: 143-154.
8. Hengfeng Wei, Yu Huang, Jian Lu, Parameterized and Runtime-Tunable Snapshot Isolation in Distributed Transactional Key-Value Stores. SRDS 2017: 21-33.
9. 易星辰, 魏恒峰, 黄宇, 乔磊, 吕建, PaxosStore 中共识算法 TPaxos 的推导、规约与精化, 软件学报, 31(8), 2020.
10. 纪业, 魏恒峰, 黄宇, 吕建, CRDT 协议的 TLA+描述与验证, 软件学报, 31(5), 2020.
11. 黄宇, 张绍文, 基于分布共享存储的移动设备数据共享方法, 中国专利, 201510205983.7, 2018-1-26.
12. 黄宇, 张昊, 基于分布共享存储的智能小车协同方法, 中国专利, 201510206612.0, 2018-7-24.