一种优化SPICE协议的探针反馈机制

程长军 刘 丹
(电子科技大学电子科学技术研究院 成都 611731)

摘 要 针对云桌面协议(Simple Protocol for Independent Computing Environments,SPICE)存在的问题:在用户交互较多的场景显示时延太大、播放视频时可能导致其它操作不能得到及时响应,在原有架构上构建了一个探针通道,根据探针采集反馈结果分别对视频和富交互场景增加了不同的可切换策略。视频场景中,探针采集了其它各个通道的丢包率和时延,综合权重计算出显示通道的带宽上限;富交互场景中,探针采集用户对设备使用的频率,在高频率时通过减少服务端推送时间间隔使显示及时刷新。在不同场景下的测试表明该方法能有效地提升视频和富交互场景下的表现。

关键词 SPICE,桌面虚拟化,探针通道,富交互
中图法分类号 TP319 **文献标识码** A **DOI** 10.11896/j.issn.1002-137X.2016.11.037

Sniffer-feedback Mechanism of Improving SPICE Protocol

CHENG Chang-jun LIU Dan
(Research Institute of Electronic Science and Technology,University of Electronic Science and Technology of China,Chengdu 611731,China)

Abstract According to the present problems of the SPICE(Simple Protocol for Independent Computing Environments):too large delay in multi-interaction scene and no response sometimes during playing video, sniff channel in origin structure was built,then some flexible policies for multi-interaction and video scenes based on feedback data from the channel were added. During playing video,SPICE server calculates a proper bandwidth from time-delay and the packet-loss-rate collected from the other channels. In multi-interaction scene,the server changes refresh interval dynamically based on the activities of input devices. The experiment results in different scenes show that it can improve the performance effectively.

Keywords SPICE,Desktop virtualization,Sniff channel,Multi-interaction

1 引言

随着互联网虚拟化概念的推进,虚拟化已经被应用到大量的科学和工程实践中[1,2],其中一项重要技术为桌面虚拟化,目前远程桌面协议有VNC,RDP,ICA和SPICE等[3,10]。SPICE是专门设计应用于虚拟环境的自适应远程传送协议,与RDP和ICA相比,其研发历史较短。利用现代高性能通信技术,SPICE旨在为带宽密集型应用提供无缝的用户体验,使虚拟桌面系统的用户体验与物理桌面系统趋于一致。从Red Hat公司开源SPICE至今[4],SPICE已经拥有越来越多的用户,并凭借着良好的虚拟桌面体验吸引了越来越多的客户。

当前主流虚拟桌面均有一定缺陷,不能完全适用于所有应用场景。而其中SPICE协议是一个开源的虚拟桌面传输协议,为用户提供与本地主机相似的交互体验,并尝试分担部分CPU密集型和GPU密集型的任务到客户端。SPICE采用独特的分层技术、分类压缩技术以及支持图像自适应处理等技术,具有图像展示体验好、对传输带宽要求较低、传输安全性高等优点。但SPICE本身也有一些不足之处,如服务端读取命令延迟、不同图像压缩算法之间切换开销大、视频处理能力不足以及用户可控性不强等。基于当前SPICE的架构和实现机制,已经有一些改进方案被提出。文献[5]提出采用JPEG2000压缩算法替换原有图形压缩算法,实现图像的渐进传输,图像和视频压缩算法之间可以平滑地切换,由此减少不同压缩算法切换导致的抖动。文献[6]利用离散小波变换对MJPEG压缩中的离散余弦变换进行替换,构建了新的视频压缩方法,用以增强SPICE的视频处理能力。然而,目前针对SPICE的改进方案大部分都是基于压缩算法来优化图形以及视频的传输机制,在操作频繁和交互性高的场景下,这类优化机制适应性差,而且对于虚拟桌面服务的可控性少有考虑。对比其它虚拟桌面协议的优化方案,比如Ctrix的ICA,文献[7]提出了3项优化方式:降低色深、输入缓冲区、减少Speedscreen延迟,此方法通过牺牲品质以换取流畅度。VNC由于客户端拉取的特殊机制,优化途径就更为简单,文献[8]在靠近服务端的位置构建了一个消息加速器作为一个代理,快速地把从服务端拿到的内容交给客户端。因此,针对

到稿日期:2015-10-08 返修日期:2016-02-26
程长军(1990—),男,硕士生,主要研究方向为信息安全,E-mail:int64ago@163.com;刘 丹(1969—),男,博士生,副教授,主要研究方向为信息安全。

其它虚拟桌面协议的优化方法很难直接应用到 SPICE 上，本文以 SPICE 原有框架为基础，在服务端和客户端之间构建了探针通道，同时针对视频传输问题和富交互场景进行优化，目的是在不损失控制流畅度的前提下最大化提高虚拟桌面的服务质量，并且通过对比实验验证了该优化机制的有效性。

2 相关工作

2.1 SPICE 图形指控流

图 1 展示了 Guest 与 Client 之间图形数据流的交互形式。其中，在 QEMU 里需要借助 libspice，这个库也可以被其它兼容 VDI[9] 的主机使用。图形指令数据流从一个需要操作系统图形引擎(X 或者 GDI)渲染操作的应用发出，图形引擎首先将指令数据流传给 QXL 驱动，QXL 驱动将操作系统的指令翻译成 QXL 指令，然后将其加入到一个指令环中，指令环是设备内存里的一个区域。Libspice 将指令从环里取出并将其加入到图形指令树里[13]，指令树包含了将要产生显示内容的指令集合。libspice 通过指令树有选择地丢弃一些指令来优化指令传递，指令树也能用来进行视频流探测。Libspice 同时还维护了一个即将发送给客户端用于呈现的指令队列，指令从队列被拉到客户端时，会被转化为 SPICE 协议消息。此时，这个指令会同时从指令树和队列里移除。当 libspice 不再需要某个指令时，指令会被放到设备释放环里，相关驱动会利用这个环释放指令资源。最后，Client 接收到图形指令时，就会更新一次显示。

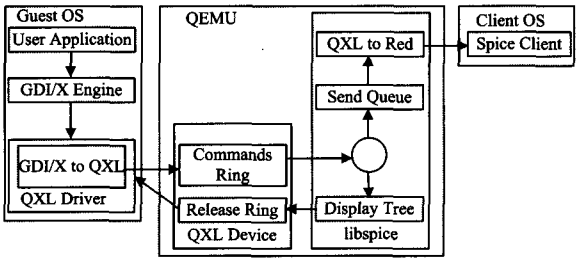

图 1 SPICE 图形指控流模块关系

2.2 SPICE 代理指令流

Spice Agent 是运行在 Guest 里的一个扩展程序模块，Spice Server 和 Client 利用代理模块执行一些必须在 Guest 里执行的任务，比如调整屏幕分辨率等。图 2 展示了 Spice Client 和 Server 通过代理的 VDI 端口驱动和 Guest 驱动进行通信的过程。消息可以由 Client 产生(设置分辨率等)，也可以由代理产生(如 ACK 等)，还可以由 Server 产生(如鼠标动作)。驱动的通信是通过其本身的输入输出环进行的，Client 和 Server 产生的消息会被写到 Server 端的队列里，然后被写入到设备输出环里，消息的读取是从设备输入环到 Server 的缓冲区。最后，消息端口决定消息是直接被 Server 处理还是被转移到 Client。

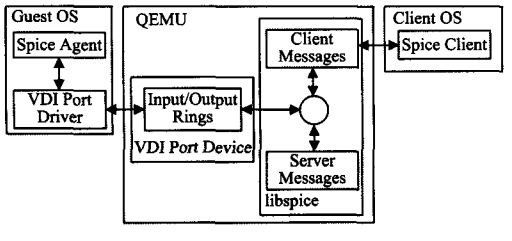

图 2 SPICE 代理指令流模块关系

3 改进方案设计

3.1 探针构建

SPICE 中，Application 是主类，它包含了 Client，Monitors 和 Screens。它主要处理通用的过程，如解析命令行参数、维护消息循环、处理事件(连接、断开连接、错误等)、切换全屏模式、将鼠标事件转化为输入事件等。

Client 与 Server 通过若干通道来进行通信，每个通道专用于传输特定的数据，如图 3 中，每个通道使用一个独立的 TCP 套接字，这个套接字可以是加密的(如 SSL)[12]，也可以是明文的。在 Client 端每一个通道都有一个独立的线程处理，因此可以根据不同服务的优先级决定不同的 QoS[11]。Server端通道同Client 端一样，包括 Main，Inputs，Display，Cursor，Playback，Record 等通道。Main 和 Inputs 受处理函数控制，Display 和 Cursor 受 Red Worker 线程控制，Playback 和 Record 有自己的处理器。

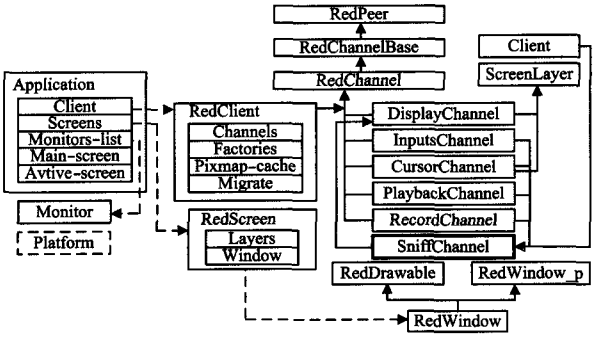

图 3 探针通道以及其它通道

在不影响原有结构的基础上，可以在 Server 和 Client 间添加一个特定的探针通道(Sniff Channel)，主要用来检测其它通道的健康状况，以及统计客户端交互的频繁程度。

3.2 富交互场景设计

SPICE 中图形是按照固定频率推送的，刷新采用懒操作机制[5]。SPICE Server 将部分显示命令进行缓存，然后当需要处理图形命令时合并发送到 Client，如图 4 所示。当 SPICE Server 尝试从指令环里读取指令时，如果环是空的，那么 Server 将回退到 epoll_wait 等待 CMD_RING_POLL_TIMEOUT(10ms)[13]，这意味着只有等到下一个 10ms 时才可能刷新显示，导致一些指令流超时太久。从以上分析可知，SPICE 在图形处理过程中注重减少带宽消耗及网络传输的数据量，但没有考虑用户在不同场景下对服务响应时间容忍度不一样的问题，忽略了服务响应的及时性，这使得 SPICE 在普通应用场景下能获得较好的用户体验，但在交互性高的应用场景中效果不佳。

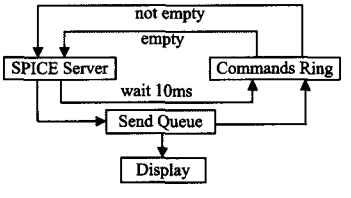

图 4 富交互场景优化前

为此本文设计了一种在富交互场景下的方案，其可保证用户在富交互应用中获得更加流畅的体验，同时在带宽不足或者交互较少时回退到默认的策略。

如图 5 所示，探针通道在 Client 端检测用户设备的使用频率，当发现用户设备使用频繁(富交互)时，设置服务端推送间隔为 2ms，从而服务端检查图形命令环的频率就会加快，使得交互可以及时在屏幕上得到响应，提高了整体的流畅度。

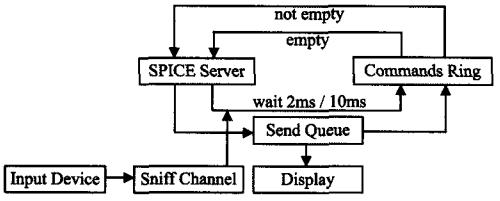

图 5　富交互场景优化后

3.3　视频场景设计

SPICE 每个通道都负责单一的数据交互，在不考虑文件传输的情况下，显示通道(Display Channel)无疑是占用带宽最多的通道，在带宽充足的情况下 SPICE Client 会获得良好的图形图像体验。但在带宽不足而播放视频时，显示通道可能耗尽了带宽，导致其它通道不能及时得到响应，出现假死现象。用最大带宽传输图像及视频且不影响其它通道传输是获得良好体验的关键，但是原 SPICE 框架并没有很好地考量显示通道应占用多少带宽。

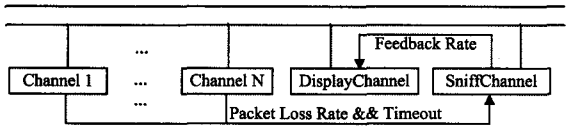

图 6　视频场景设计

针对视频这个特殊场景，参考了 TCP 中的拥塞避免算法[13]，借助探针反馈思想设计了一套算法。如图 6 所示，在 Server 与 Client 之间增加了一个新的通道作为探针(Sniff Channel)，探针采集除了显示通道以外通道的时延和丢包率，最终评估出显示通道该占用的带宽。如果带宽不足以支撑当前传输的图像视频，可以参考文献[15]，采用丢帧的策略进行进一步优化。考虑一般场景，通道 i 的丢包率 $r_i(t)$ 和时延 $s_i(t)$ 跟通道带宽 $w_i(t)$ 的关系可粗略表示为：

$$r_i(t) \propto 1/w_i(t) \tag{1}$$

$$s_i(t) \propto 1/w_i(t) \tag{2}$$

$$ar_i(t)+bs_i(t) \propto 1/w_i(t) \tag{3}$$

其中，a/b 分别为丢包率/时延的权重。不失一般性，假设 $w_0(t)$ 是显示通道带宽，$w_1(t)-w_N(t)$ 是其它通道带宽，则有：

$$w_0(t)=BW(t)-\sum_{i=1}^{N} w_i(t) \tag{4}$$

$$w_0(t)=BW(t)-\sum_{i=1}^{N} \frac{p_i}{ar_i(t)+bs_i(t)} \tag{5}$$

其中，p_i 为通道 i 的权重，$w_0(t)$ 是显示通道受影响的期望带宽，也可以近似当作其它通道的拥塞程度，$BW(t)$ 表示总带宽，大部分情况下为一个固定值。

为了在播放视频时动态调节显示通道带宽，使得其它通道的交互的影响程度最小，可以使用调节率 CR 作为显示通道带宽调节的参考指标：

$$CR=\frac{w_0^2-w_0^1}{\Delta w} \tag{6}$$

其中，Δw 为显示通道带宽调节量；$w_0^2-w_0^1$ 为 $w_0(t)$ 带宽调节前后的变化量；$CR \to 0$ 说明调节是没必要的，表明其它通道处于流畅状态；$CR \to 1$ 说明调节是非常必要的，表明此时其它通道很可能处于拥塞状态。CR 临界值的选取需要根据环境实际情况测定，在本文测试环境下 CR 临界值取0.4。

因为所有通道共用一个物理通道，所以显示通道的带宽调节也会影响其它通道，其它通道又会经过探针通道的采集计算得到新的显示通道带宽，这最终会达到一个平衡状态，即在不影响其它通道交互的前提下，得到了当前最优的显示效果。

4　实验评估

本文对提出的探针机制进行了测试，并在一定条件下进行了比较。富交互和视频场景中，带宽都是一个较大的影响因子，所以实验是在不同网络环境下进行的。实验所用的 SPICE Server 配置：Intel(R) Xeon(R) E5-2620 v2 @ 2.1HG x 24、64GB 内存、Broadcom BCM5719 千兆网卡，Client 为 Thinkpad T410s 型号的笔记本，通过 TP-Link TL-SF1024S 交换机连在同一网段，带宽调节通过 Netlimiter 控制，Client 和 Guest 操作系统均为 32 位 Windows7。评估 QoE 的方法较多[16]，这里使用通道平均时延作为衡量标准。

4.1　富交互对比测试

SPICE 虚拟桌面对鼠标键盘的传输和响应不同于操作系统内核本身的机制[17]，其走的是单独的 Inputs 通道，指令到达 Server 后再通过 VDI 传输到 Guest，所以若要统计交互频率，只需在此通道后端加一个计数器来统计鼠标键盘事件数(MKA)。本次实验选择在较普遍的 3 种场景(见表 1)下进行测试，并分别在 Client 上设置了 3 种带宽：50kB/s，500kB/s，5000kB/s。

表 1　3 种场景对应的鼠标键盘事件频率

场景	基本无动作	普通阅读或上网	频繁办公和窗口切换
鼠标键盘事件(MKA/min)	0—5	40—70	150—330

如图 7 所示，在带宽充足时优化的作用并不明显，而且对于不同使用场景也没有太大的区别，均可流畅使用。在带宽不足时优化效果比较显著，性能甚至会提高一倍；对比不同的场景，在交互比较多的场景下优化后的 SPICE 协议充分发挥了作用。综上所述，优化后的协议比较适合交互较多和带宽不充裕的场景。

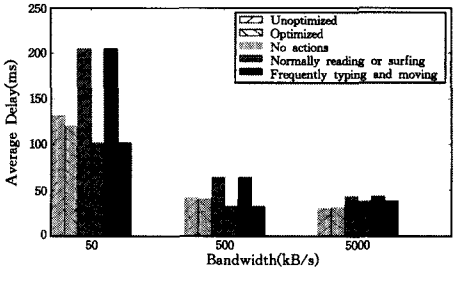

图 7　富交互场景优化前后时延对比

4.2　视频场景对比测试

在 Guest 里播放 720P 的视频，分别在优化和未优化的情况下测试随着带宽增加交互所产生的各个通道的平均时延。

如图 8 所示，因为视频占用的带宽较多，所以在带宽不足时会出现整体无法响应的现象，之后随着带宽增加，优化后的

(下转第 209 页)

[10] Karuppiah M, Saravan R. A secure remote user mutual authentication scheme using smart cards[J]. Journal of Information Security and Application, 2014, 19(4): 282-294

[11] Gan Hong, Pan Dan. Research on dynamic ID-based remote user authentication scheme[J]. Digital Communication, 2014, 41(5): 1-5(in Chinese)
甘宏，潘丹. 基于动态身份远程用户认证方案的研究[J]. 数字通信，2014，41(5)：1-5

[12] Liu J Y, Zhou A M, Gao M X. A new mutual authentication scheme based on nonce and smart cards[J]. Computer Communications, 2008, 31(10): 2205-2209

[13] Huang H F, Liu S E, Chen H F. Designing a new mutual authentication scheme based on nonce and smart cards[C]// International Symposium on Parallel and Distributed Processing with Application, ISPA 2010. Taipei, Taiwan, 2010: 570-573

[14] Li C T, Hwang M S. A lightweight anonymous routing protocol without public key en/decryption for wireless ad hoc network [J]. Information Science, 2011, 181(23): 5333-5347

(上接第 192 页)

SPICE 客户端的时延会迅速趋于稳定，而未优化的客户端的时延趋于稳定的时间比较缓慢，在带宽充足(大于 5000kB/s)后双方差异越来越不明显。由此可见，优化后的客户端在普通带宽下播放视频表现会比较出色，这也是大多数的应用场景。

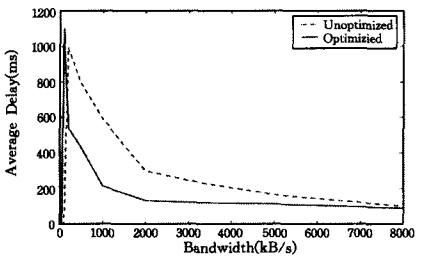

图 8 视频场景优化前后时延对比

结束语 通过分析 SPICE 机制以及某些场景下的特殊性，本着在不影响使用的前提下最大限度提高流畅度的原则，提出了在 SPICE 服务端和客户端之间构建探针通道，用以收集反馈信息，分别针对富交互和视频场景设计了不同的切换策略。实验表明，此方法可以在大多数情况下提升 SPICE 虚拟桌面在富交互和视频场景中的表现。

参 考 文 献

[1] Yu Tao, Bi Jun, Wu Jian-pin. Research on the Virtualization of Future Internet[J]. Journal of Computer Research and Development, 2015, 52(9): 2069-2082(in Chinese)
余涛，毕军，吴建平. 未来互联网虚拟化研究[J]. 计算机研究与发展，2015，52(9)：2069-2082

[2] Wu Jie, Kan Wen-xiao, Du Ran, et al. Application-oriented Virtualization Desktop Grid Architecture[J]. Computer Engineering, 2014, 40(3): 88-92(in Chinese)
武杰，阚文枭，杜然，等. 面向应用的虚拟桌面网格架构[J]. 计算机工程，2014，40(3)：88-92

[3] Zhang Jian-xun, Gu Zhi-ming, Zheng Chao, et al. Review of Cloud Computing Research[J]. Application Research of Computers, 2010, 27(2): 429-433(in Chinese)
张建勋，古志民，郑超，等. 云计算研究进展综述[J]. 计算机应用研究，2010，27(2)：429-433

[4] Simoens, Pieter, et al. Remote display solutions for mobile cloud computing[J]. Computer, 2010, 8: 46-53

[5] Xu Hao, Lan Yu-qing. Research on desktop virtualization technology based on SPICE protocol and its improvement solutions [J]. Computer Engineering & Science, 2013, 35(12): 20-25(in Chinese)
徐浩，兰雨晴. 基于 SPICE 协议的桌面虚拟化技术研究与改进方案[J]. 计算机工程与科学，2013，35(12)：20-25

[6] Qiao Yong. Research and Improvement of Video Transmission of SPICE in Cloud Desktop Environment[D]. Jinan: Shangdong Uiniversity, 2013(in Chinese)
乔咏. SPICE 协议的视频传输分析与改进[D]. 济南：山东大学，2013

[7] Schlosser, Daniel, et al. Improving the QoE of citrix thin client users[C]// 2010 IEEE International Conference on Communications(ICC). IEEE, 2010: 1-6

[8] Taylor, Cynthia, Pasquale J. Improving video performance in vnc under high latency conditions[C]// 2010 International Symposium on Collaborative Technologies and Systems(CTS). IEEE, 2010

[9] Massie, Thomas H, Salisbury J K. The phantom haptic interface: A device for probing virtual objects[C]// Proceedings of the ASME Winter Annual Meeting, Symposium on Haptic Interfaces for Virtual Environment and Teleoperator Systems. 1994: 295-300

[10] Richardson, Tristan, et al. Virtual network computing[J]. Internet Computing, IEEE, 1998, 2(1): 33-38

[11] Orlov, Zigmund, Necker M C. Enhancement of video streaming QoS with active buffer management in wireless environments [OL]. http://oa.sloc.com.cn/handle/0/25468

[12] BhiogadeM S. Secure socket layer [J]. Ssrn Electronics Journal, 2001, 12(4): 475-490

[13] SPICE. Improving Spice performance: problematic scenarios and ideas [EB / OL]. (2015-09-10). http://www.spice-space.org/page/EnhancePerf

[14] Allman, Mark, Paxson V, et al. TCP congestion control [J]. ACM Computer Communications Review, 1999, 29(5): 308-309

[15] Chen Dan-wei, Zhang Zeng. A Feedback Frame-dropping Strategy of Improving SPICE Virtual Video Performance[J]. Computer Technology and Development, 2014, 14(1): 136-139(in Chinese)
陈丹伟，张增. 一种提高 SPICE 虚拟视频性能的反馈丢帧策略[J]. 计算机技术与发展，2014，14(1)：136-139

[16] Kim H J, et al. The QoE evaluation method through the QoS-QoE correlation model[C]// Networked Computing and Advanced Information Management. 2008

[17] Feng Jian, Ni Ming, Guo Zi-gang, et al. Design and Implementation of Cloud-desktop Based on Video Stream[J]. Computer Enginee-ring, 2013, 39(10): 37-41(in Chinese)
冯健，倪明，郭自刚，等. 基于视频流的云桌面设计与实现[J]. 计算机工程，2013，39(10)：37-41