RED HAT :: CHICAGO :: 2009
SUMMIT

SPICE: An Open Remote Computing Solution

Presenter: Arnon Gilboa

Project Leader: Yaniv Kamay

Red Hat Israel

September 4, 2009

Agenda

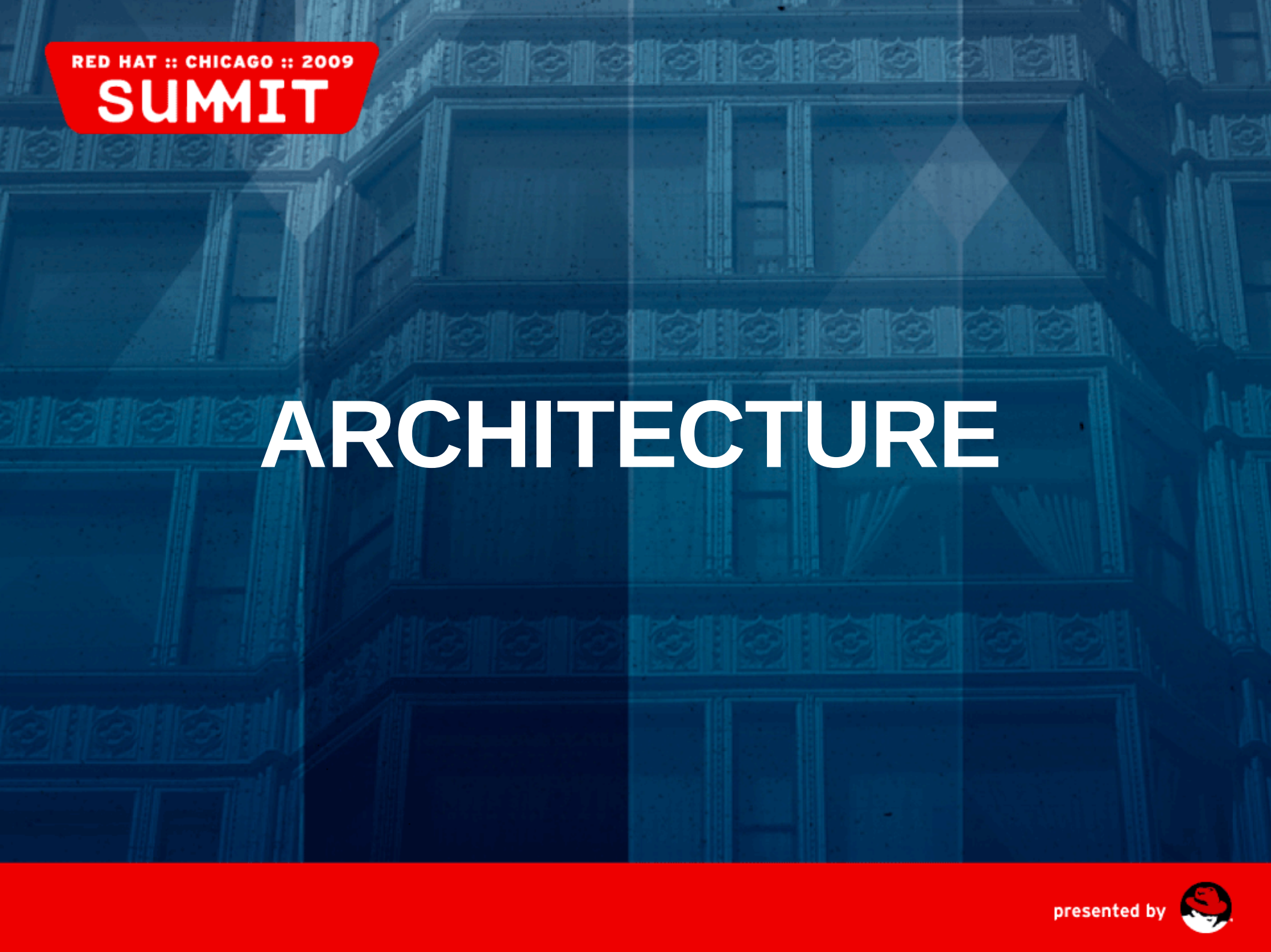
- Why Spice?
- Architecture
- Spice Optimizations
- Future Developments
- Q&A

Why Spice?

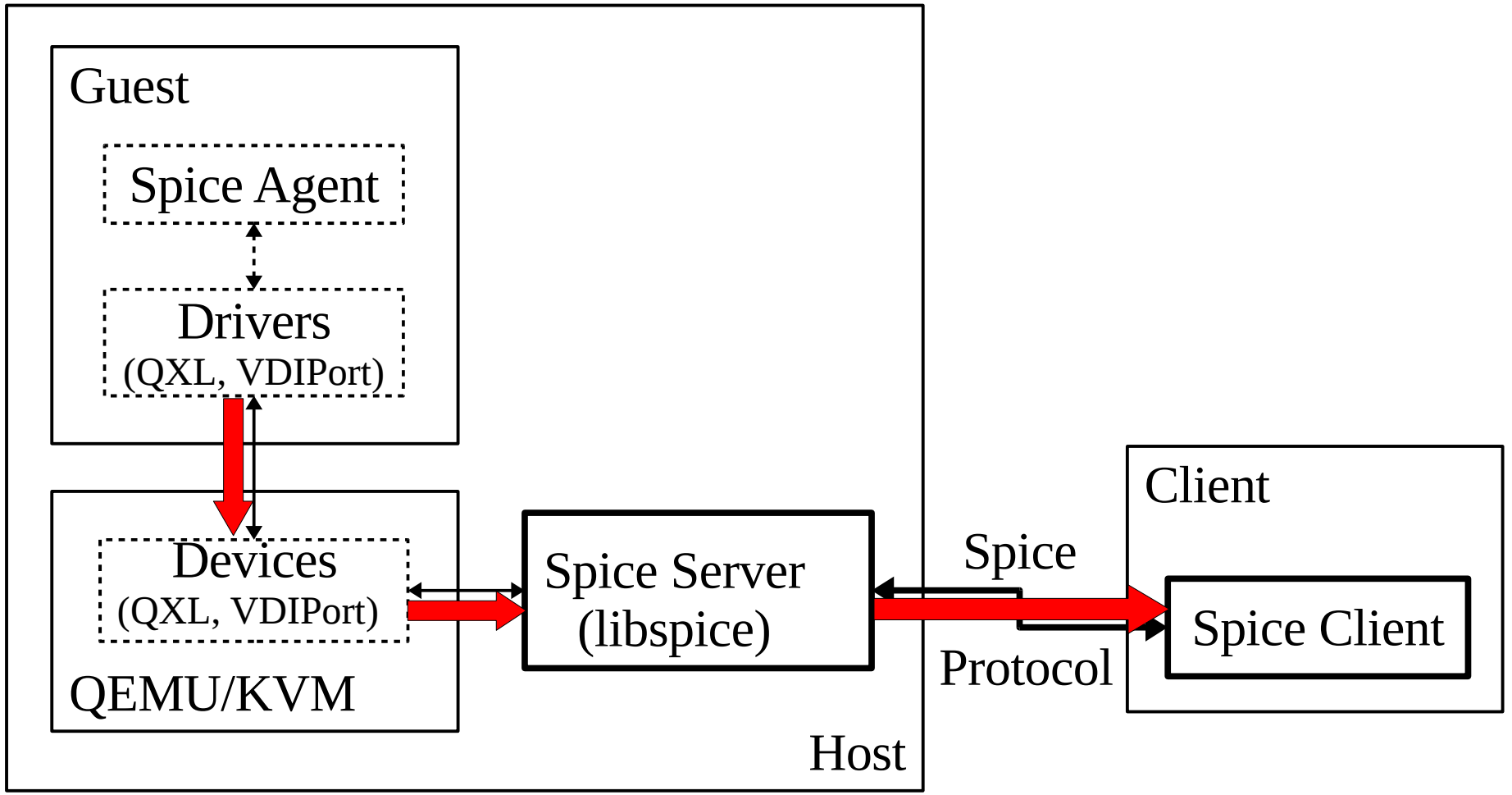
- Open remote computing solution
- Client access to remote machine display & devices
- Emphasize on virtual machines environment
  - No need for guest networking
- User experience similar to local machine
  - High quality video & audio
- Offload intensive CPU & GPU tasks to client
- Suitable for both LAN & WAN usage

How Spice?

- Transfer graphic commands/objects (currently 2D)
  - Other solutions use frame buffer updates
  - Fill, copy, stretch, path stroke, text, pointer shape etc.
- Generic & platform-independent
- Lossless images, lossy video & audio

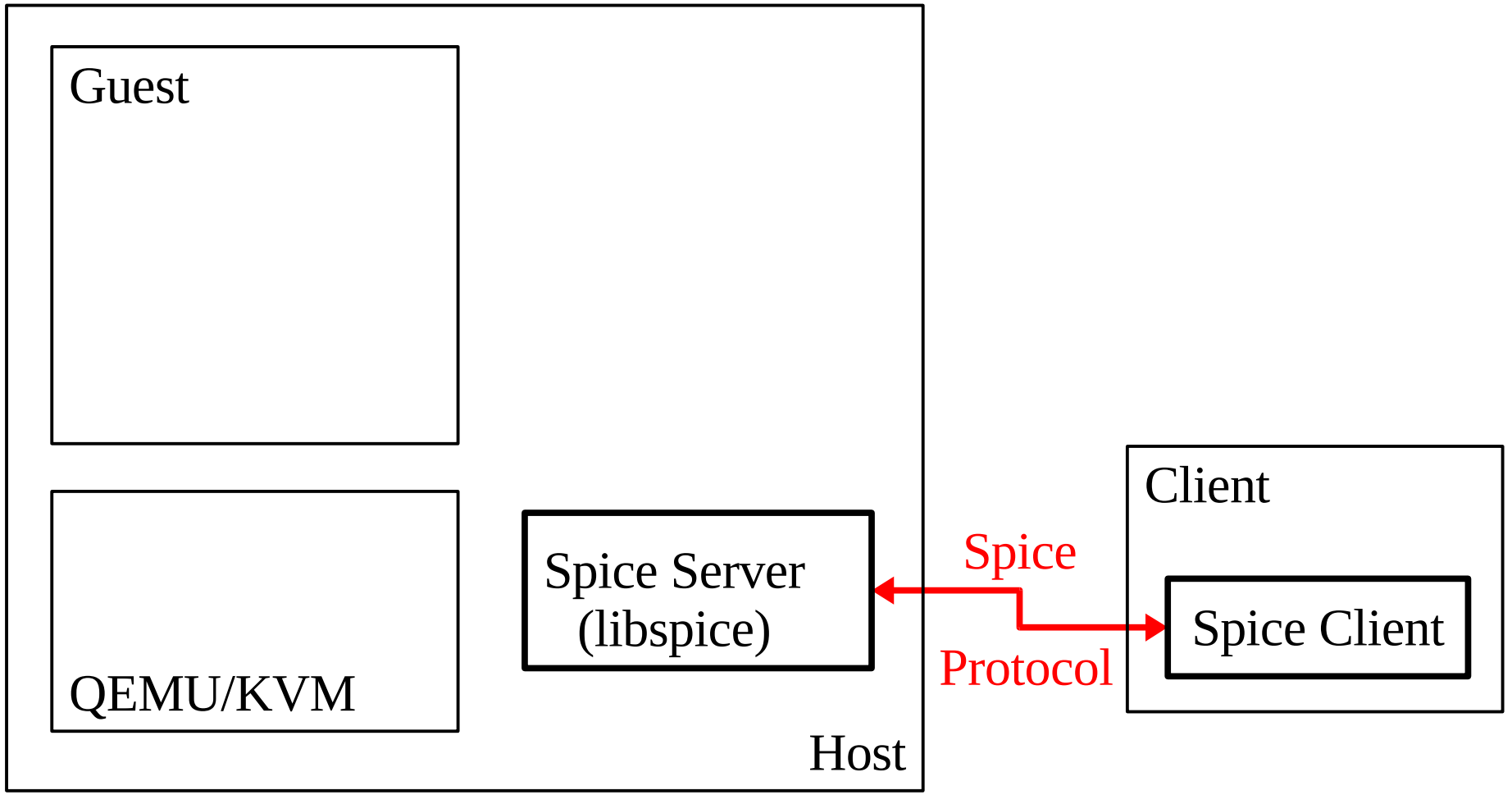

RED HAT :: CHICAGO :: 2009
SUMMIT
ARCHITECTURE
presented by

Spice Architecture



RED HAT :: CHICAGO :: 2009
SUMMIT

Spice Protocol



Spice Protocol

- Client-server communication
- Set of messages for interaction with remote devices
  - Graphic commands, keyboard & mouse events, audio streams etc.
- Secure authentication – ticketing (OTP) with expiration
- Connection establishment and control (e.g., migration)
- Flow control using window-acks

Client-Server Communication: Channels

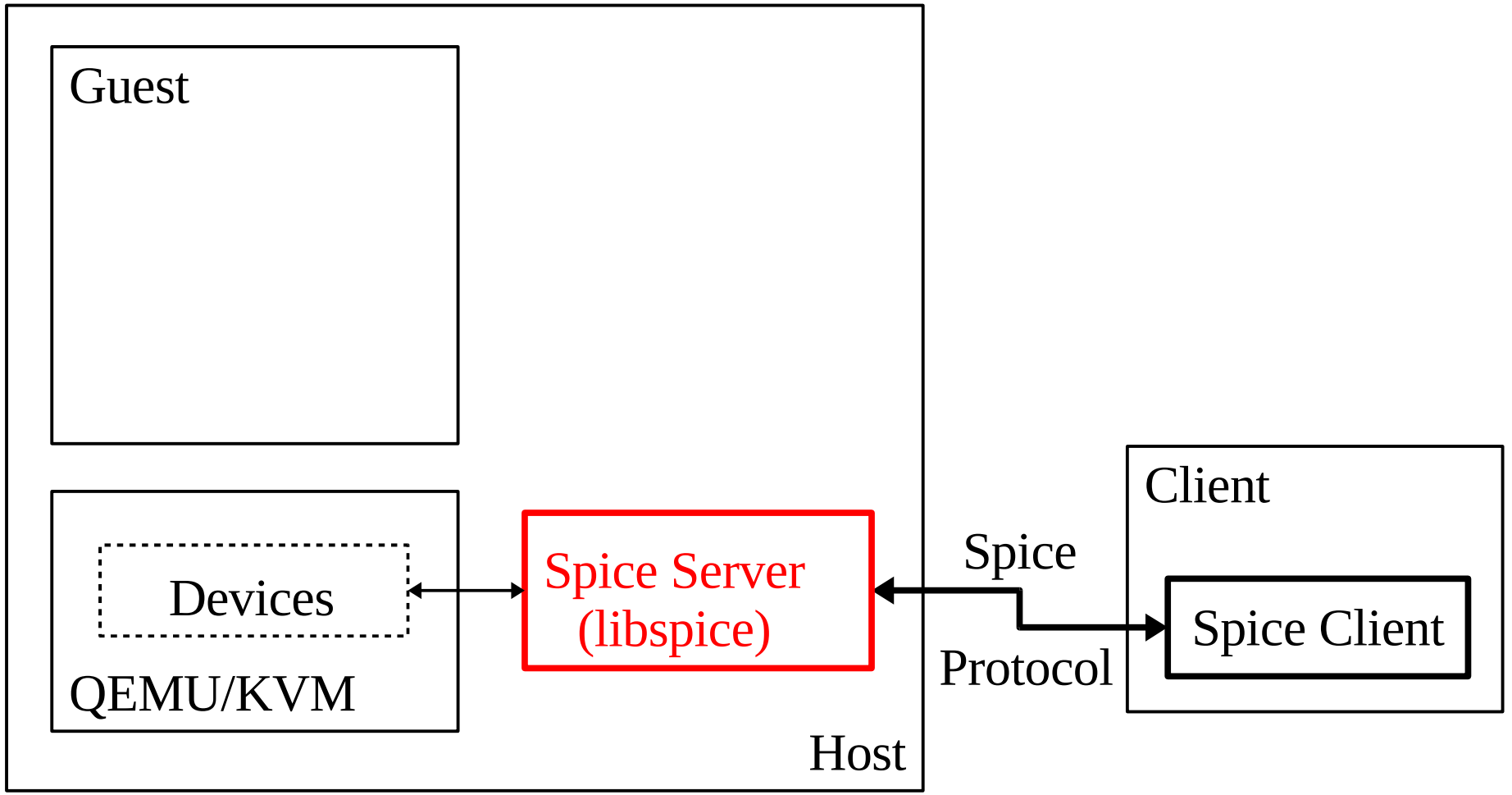
- Client & server communicate via channels
  - TCP connections
  - Each Channel dedicated to a specific type of data
  - Channel can be secured using SSL or unsecured

- In the client, each channel has a dedicated thread
  - Different QoS can be given per channel

Channels

- **Main** - control and configuration and agent communication
- **Display** - handles graphic commands, images and video streams
- **Inputs** - keyboard and mouse inputs
- **Cursor** - pointer device position, visibility and cursor shape
- **Playback** - audio received from the server to be played by the client
- **Record** - audio capture on the client side

Spice Server

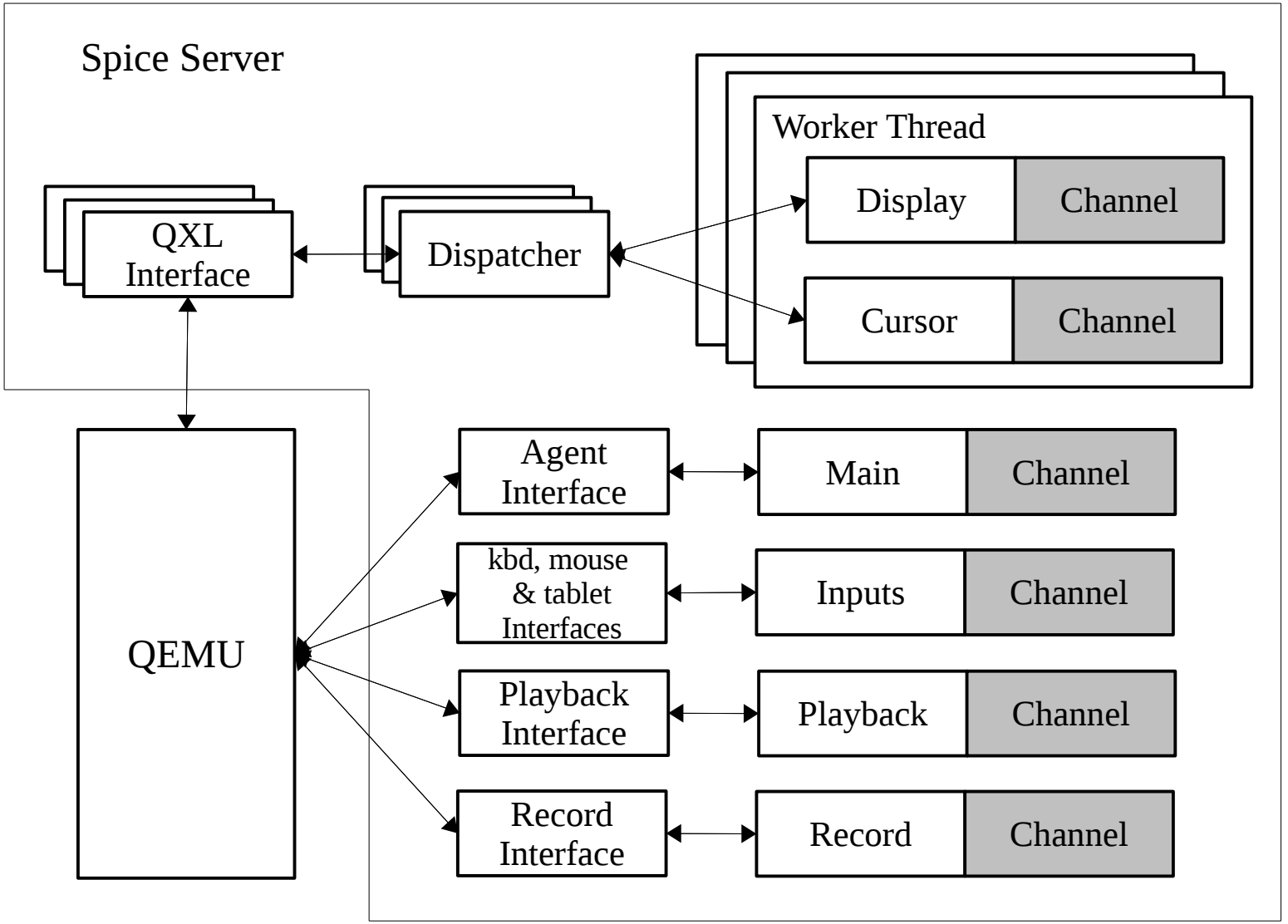


Spice Server – a pluggable library

- Virtual Device Interface (VDI)
  - Defines a set of interfaces
  - Publishes virtual devices (e.g., display, keyboard, mouse)
  - Enables interaction with devices

- libspice – the server, a VDI-pluggable library
- Interacts with VDI host application (e.g., QEMU)
- Communicates with client using Spice protocol

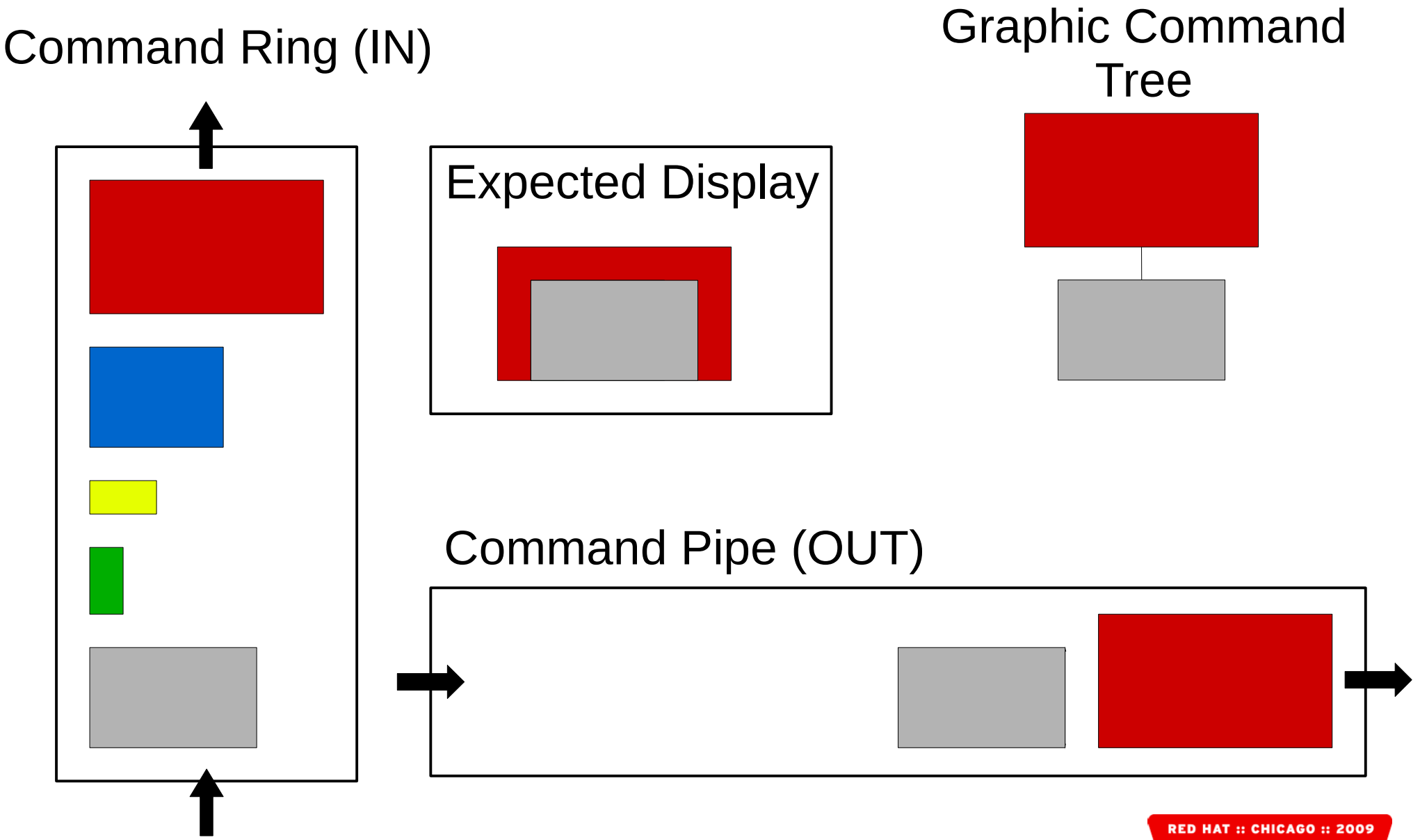
Spice Server – Interfaces



Spice Server – Graphic Commands Processing (1)

- Graphic Commands Tree
  - Contains the set of commands, whose execution will reproduce the display content
  - Used to optimize commands transmission to the client by dropping commands hidden by others
- Command Pipe
  - Commands to be sent to the client, for updating its display

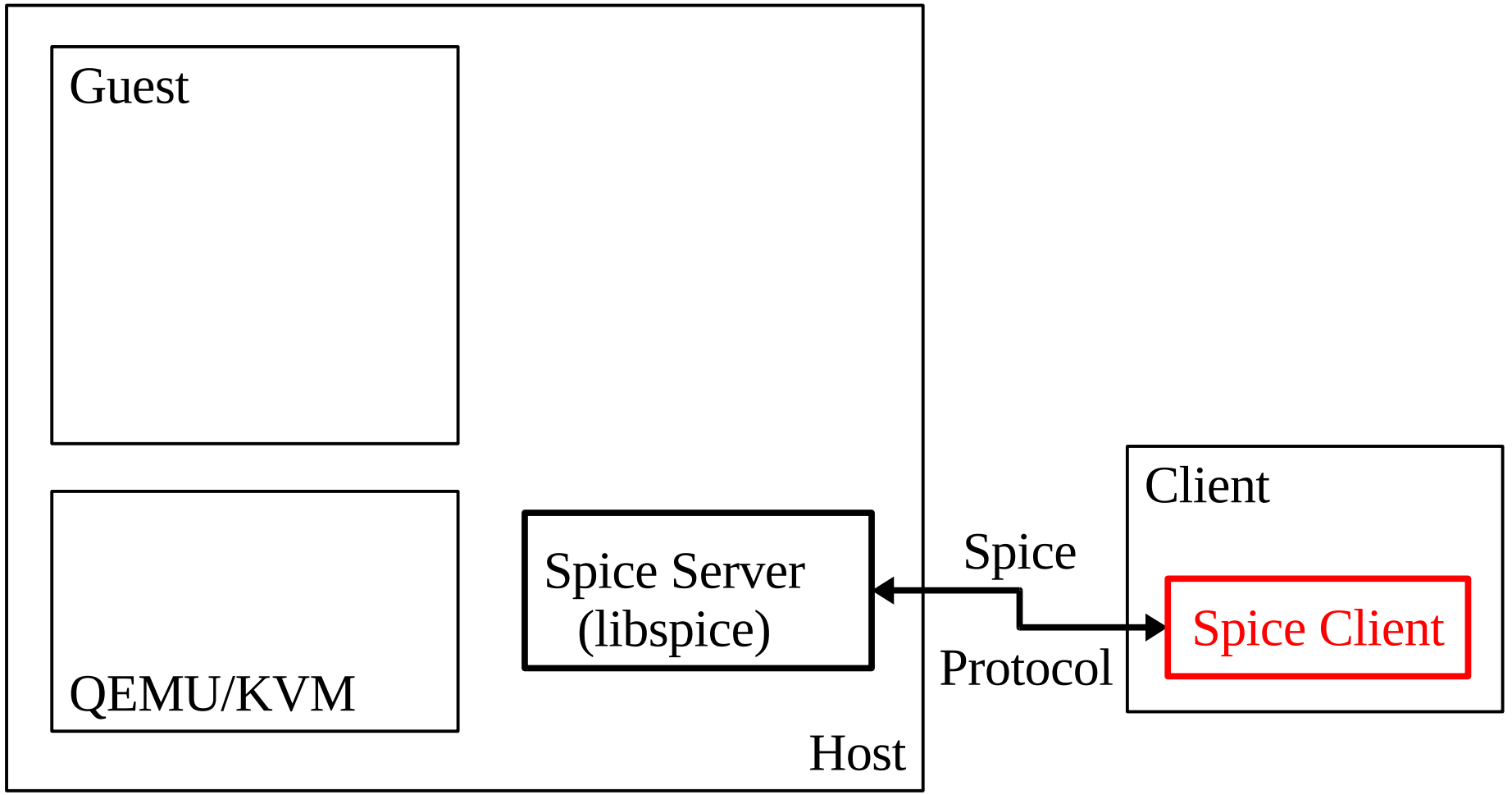
Spice Server – Graphic Commands Processing (2)



Spice Server – Graphic Commands Processing (3)

- Attempt to pass the rendering job to the client, leveraging its hardware acceleration abilities
- Rendering on the host side by software or GPU is done as a last resort
- Release a command only when
  - Completely covered by other commands and there are no dependencies on it
  - Running out of resources, the command is rendered to the frame buffer
- When the guest needs to read from the frame buffer
  - Command rendered to frame buffer & deleted from tree

Spice Client

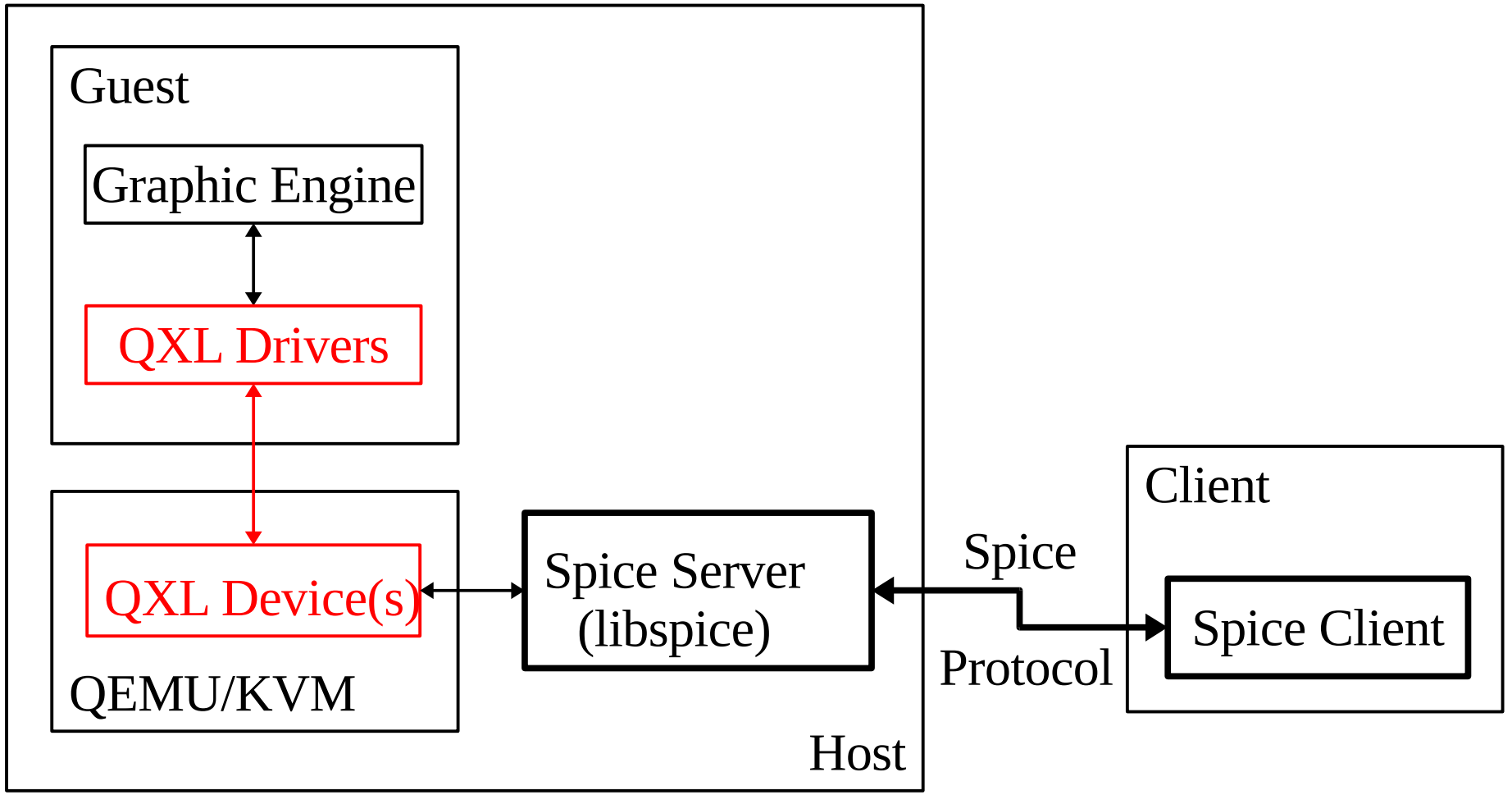


Spice Client

- End-user interface
- Cross-platform
  - Linux & Windows
  - Generic classes & interfaces
  - Parallel implementation of platform-specific areas
- Does not require any codecs


RED HAT :: CHICAGO :: 2009
SUMMIT
SPICE FRIENDS &
FAMILY
presented by

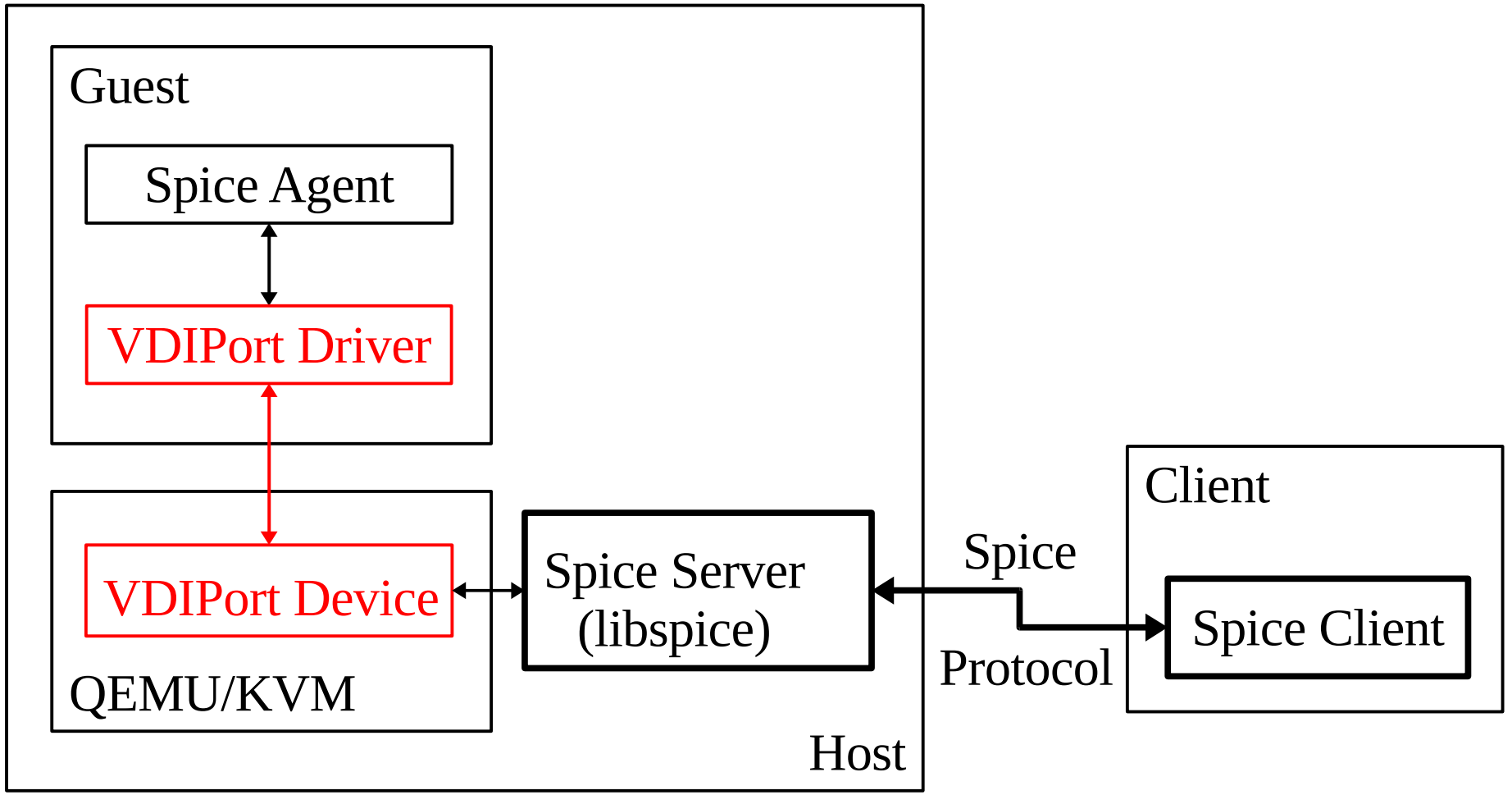
QXL Device & Driver



QXL Device & Driver

- Transparent remote display
  - Specific QEMU PCI display device
  - Paravirtual device for enhanced performance
    - Transfer generic graphic commands to server
    - Pixmap hashing & cache hint
    - Cursor support
- Windows driver & beta X11 driver already implemented
- Support standard VGA mode as well
  - No drivers required
  - Boot stage display

VDIPort Device & Driver

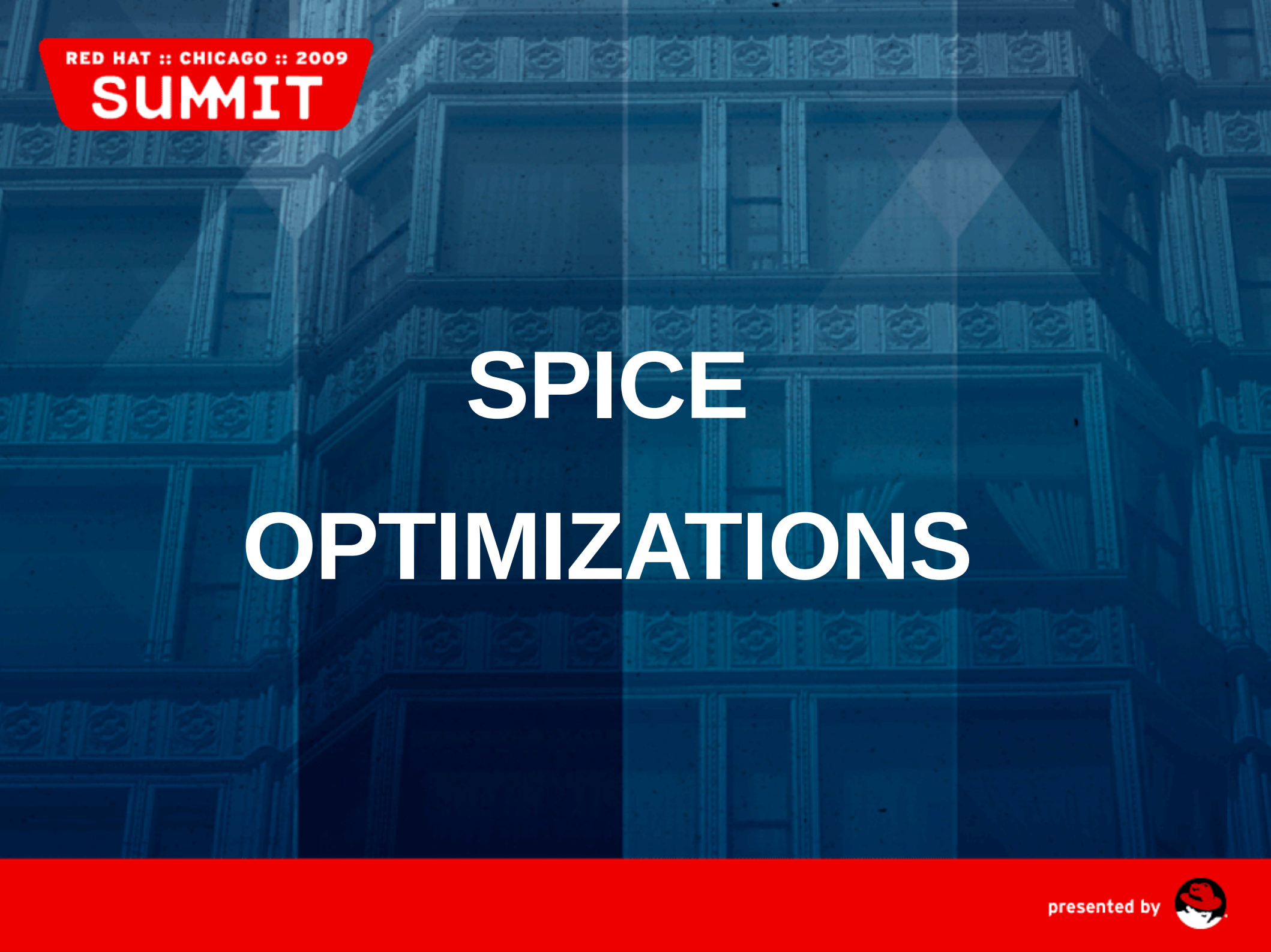


VDIPort Device

- VDI port - QEMU PCI device used for communication with guest agent
- Agent communicates with both client & server using agent-dedicated protocol
- Windows guest driver already implemented

Spice Agent

- Optional component
- Enhances user experience
- Performs guest-oriented tasks
  - Injects mouse position and state
  - Display settings configuration



SPICE OPTIMIZATIONS

Image Caching

- Client image caching: pixmaps, palettes and cursors
- Eliminates sending redundant instances of an image
- Image arrives from driver with unique id & cache hint
- Pixmap cache is shared among all the displays
- Synchronized between the server and client
- Server manages cache

Image Compression

- Lossless - commands dependency, prevents artifacts
- Image-based
  - Quic - based on SFALIC
  - LZ - LZSS adjusted to images
- Global LZ (GLZ)
  - LZ with a history-based global dictionary
  - Takes advantage of repeating patterns among images
- Auto - Heuristic compression choice per image
  - Synthetic images - LZ/GLZ
  - Photos - Quic

Video Compression

- Server heuristically identifies video streams
  - Using the commands tree
  - Areas constantly updated for a period of time
  - Send them as video stream coded using M-JPEG
- Saves a lot of traffic, improving Spice performance, especially in WAN
- Reduces client & server CPU consumption
- Does not require any codecs

Hardware Acceleration

- Basic client rendering - using Cairo
  - Cross-platform, device-independent 2D library
- Client HW acceleration
  - GDI in Windows
  - OpenGL in Linux (experimental)
- Rendering by the client GPU instead of CPU
  - High performance rendering
  - Better client CPU utilization
- Server HW acceleration uses OpenGL
  - Sharing code with the Linux client

Mouse modes

- Client mode
  - Client mouse used as the effective pointing device
  - Guest agent used for cursor position injection
  - Cursor shape & visibility - updated in a dedicated channel
  - Smooth cursor motion and responsiveness
  - Appropriate for WAN
- Server mode
  - Server controls mouse position
  - Always synchronized

Other Features

- Multiple Monitors - tested on 4, no real limit
  - Automatic configuration of guest displays
- 2-way Audio and Lip-sync
  - Audio playback and recording
  - Playback compressed using CELT
  - Lip-sync using video frames time-stamping
- Hardware Cursor
  - Separated from display, reduces network traffic
  - Enables prioritizing for better responsiveness
- Live Migration - seamless to a connected client

FUTURE DEVELOPMENTS

Future Features

- Off-screen surfaces
- Direct Draw
- Video acceleration (DXVA)
- 3D acceleration
- Aero (new drivers model)
- Network tunneling
  - Client network printer redirection
  - Generic Resource sharing

Future Features - cont.

- Client GUI, preferences and window management
- USB device redirection
- Clipboard sharing
- CD-ROM redirection

QUESTIONS?

Additional Information

- http://www.spice-space.org (soon to come...)
- Spice documentation effort
  - User manual
  - Spice remote computing protocol definition
  - Spice VD Interfaces documentation
- spice-space-list
- Contact:
  - agilboa@redhat.com