



# VIRTLAB Project

## Virtual Networking Laboratory

**VSB-Technical University of Ostrava  
Faculty of Electrical Engineering and Computer Science  
Department of Computer Science  
Regional Cisco Networking Academy**



**Project Leader: Petr Grygárek**



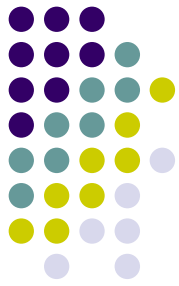
# Pedagogical Goals

- Give students more opportunity to exercise and make their own experiments with networking laboratory equipment
  - laboratory is occupied in working hours
- Allow distant-learning students to do practice laboratory exercises
- Utilize costly laboratory devices more efficiently
  - allow remote access during non-working hours
- Share special and/or expensive devices between universities

# Why Not to Use Commercial Solutions ?

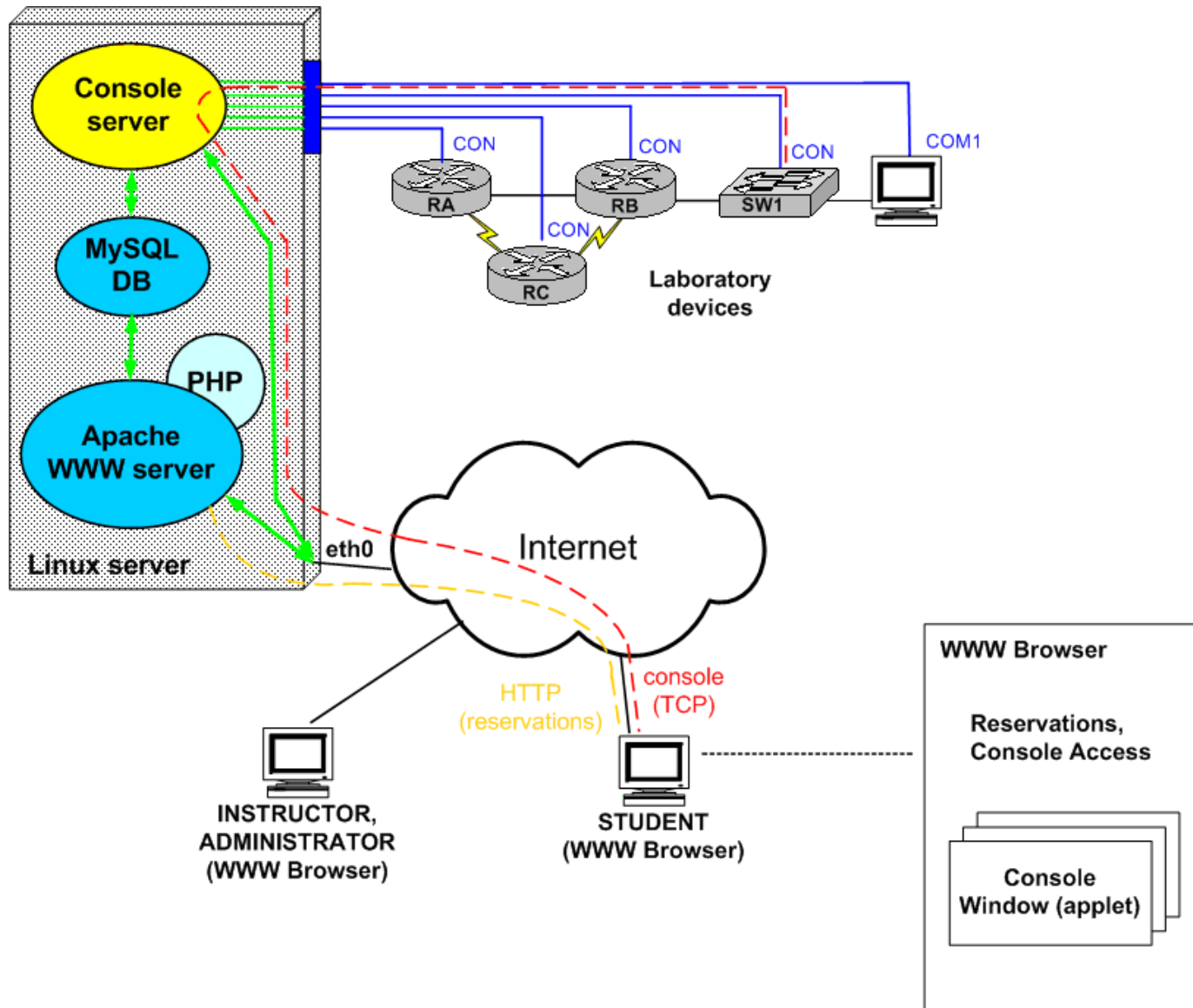


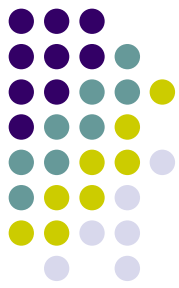
- Remote access solutions using hardware-based terminal servers were implemented by many universities
  - expensive
  - provides little flexibility - low potential for advanced feature implementation
  - often lacks access management system at all
- Software like Netlab requires costly periodic license renewal and can not be extended to implement features user-specific required
- We wanted to develop task-based, multiple-site variable-topology solution



# The Original Single-Site Architecture

# The Original Idea and Architecture





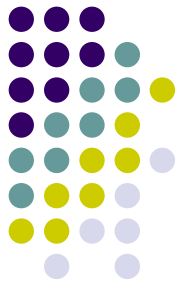
# Basic Goals

- Make consoles of lab devices accessible remotely
- Create access management system to securely share devices among students
- Provide a set of tutor-defined tasks
  - students commonly don't know what exactly they could to experiment with
- Be able to handle various network topologies
  - fixed topology common for all tasks proved limiting

# Implementation Platform



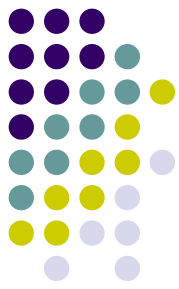
- Open-source technologies
  - Apache+PHP, ANSI C/C++, MySQL, XML, Java applets
- Server side runs on Linux
  - Transfer to other platforms possible, but not expected
- On the client side, only standard WWW browser is required
  - Java applet support is needed (Java 1.4)



# Access Management System



# Access Management System Philosophy



- Task-oriented device access
- Task defined by objectives, devices the student may access and network device topology
- Particular tasks are offered to students in timeslots chosen by administrator using electronic notice-board
  - Division of time into timeslots advantageous if task topologies are different and have to be connected manually
- Students may reserve timeslots to solve particular tasks
  - up to student's weekly quota



# Tasks

- Student may choose from extensible set of task created by tutors
- Special case of task: student may define his/her own topology
- Task completely described using XML/HTML
  - objectives may include not only text and topology picture but also additional multimedia elements
- Multiple students may cooperate on task solution
- Task solutions (correct configuration files) may be made available to students

# Roles of Access Management System Users



- Task Creator (commonly instructor)
- Task Scheduler (administrator)
- Student
- System Administrator
  - administers user, devices, adjusts system parameters
- Tutor

An user may have multiple roles simultaneously.



# Task Creator Role

- Creates, edits and deletes tasks
  1. using Web form
  2. Offline creation and insertion of archive with prescribed contents
    - HTML pages + XML-based description
- tasks may be organized using user-definable category system

# Task Creation - Screenshot



Virtuální síťová laboratoř - Mozilla Firefox

File Edit View Go Bookmarks Tools Help

http://localhost/web\_lin456/vytvor\_ulohu.php?si=1363752761&count=1

Red Hat, Inc. Red Hat Network Support Shop Products Training

SMS.cz - děláme to lépe Virtuální síťová laboratoř

## VYTVOŘENÍ ÚLOHY

Nazev ulohy:

Max. pocet resitelu :

Pocet hodin pro řešení :

Obrazek:

Databasmekek konf. soubor:

### Popis ulohy

```
<h1>Konfigurace protokolu RIP </h1>
<h2>Zadani</h2>
<p>
Zapojte ulohu dle schematu.
Vymyslete oadrosovani site a nakonfigurujte zarizeni
otestujte zapojeni pomoci prikazu ping
</p>
<h2>Poznamka</h2>
<p>
dejte pozor na nastaveni Clock rate na zarizeni DCE,
pri problemech nejdrive zkontrolvat, zda neni interface shutdown.
</p>
```

nacist ze souboru:

**Úlohy**

- Hlavní
- vytvoření
- procházení / edit
- kategorie / edit
- zpřístupnění

**Zařízení**

- vytvoření
- procházení / edit

**Uživatelé**

- administrace

**Ostatní**

- Statistika

**Odhlášení**

- Odhlášení

Smerovace	Prepinace	Rozbocovace	Ostatni
<input checked="" type="checkbox"/> RA	<input type="checkbox"/> SW1	<input type="checkbox"/> H1	
<input checked="" type="checkbox"/> RC	<input checked="" type="checkbox"/> SW2	<input checked="" type="checkbox"/> H2	
<input checked="" type="checkbox"/> RD	<input checked="" type="checkbox"/> SW3	<input type="checkbox"/> H3	
<input type="checkbox"/> RF	<input type="checkbox"/> SWITCH 5		
<input type="checkbox"/> RT			

**Kategorie**

[Pridat Kategorii](#)

Done

pavel@pavel1:/home/pavel pavel@pavel1:~/skola/diplomka Virtuální síťová laboratoř - Mozill vytvor\_ulohu.php (/var/www/htm

Applications Actions

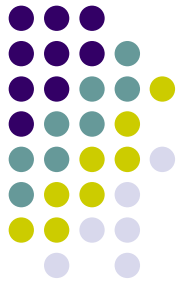
Fri Apr 1, 01:07



# Task Scheduler Role

- Defines which task is accessible in particular timeslot
- Task schedule is available to students on electronic notice-board
  - Students may reserve particular timeslots
- Time divided to fixed-size timeslots (45 mins)

# Task Scheduling - Screenshot



Virtuální síťová laboratoř - Firefox

Soubor Úpravy Zobrazit Přejít Záložky Nástroje nápověda

http://linux456.vsb.cz/~nem114/diplomka/zpristupneni\_ulohy.php?si=619475238

Mozilla Firefox Přehled zpráv

### April, 2005

<< Prev week 14 Next week >>

Úloha: konfigurace OSPF

	0:00	0:45	1:30	2:15	3:00	3:45	4:30	5:15	6:00	6:45	7:30	8:15	9:00	9:45	10:30	11:15	12:00	12:45	13:30	14:15	15:00	15:45	16:30	17:15	
4 Pondělí																									
5 Úterý	OSPF-1	OSPF-1 RIP-1	OSPF-1 RIP-1	OSPF-1 RIP-1	OSPF-1 RIP-1	OSPF-1									RIP-1 OSPF-1	RIP-1 OSPF-1	RIP-1 OSPF-1								
6 Středa	POK	POK	POK			RIP-1	RIP-1	RIP-1	RIP-1	RIP-1					OSPF-1	OSPF-1 RIP-1	OSPF-1 RIP-1	OSPF-1	OSPF-1	OSPF-1	OSPF-1	OSPF-1	OSPF-1	OSPF-1	OSPF-1
7 Čtvrtek	OSPF-1	OSPF-1	OSPF-1	OSPF-1	OSPF-1								OSPF-1	OSPF-1	OSPF-1	OSPF-1	OSPF-1	OSPF-1	OSPF-1	OSPF-1	OSPF-1	OSPF-1	OSPF-1	POK	POK
8 Pátek	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
9 Sobota	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	OSPF-1 RIP-1	OSPF-1 RIP-1	OSPF-1 RIP-1	OSPF-1 RIP-1	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
10 Neděle	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>

[\[zpět\]](#)

Hotovo

pavel K3b - Virtuá

13:39

# Student Role



- Reserves timeslots offered using electronic notice-board for himself/herself
  - may also lists colleagues who he/she wants to solve the task with him/her
- Can access devices' consoles of reserved task in his/her timeslot using Java applet running in his/her WWW browser



# Remote Device Access - Screenshot



Virtuální síťová laboratoř - Firefox

Soubor Úpravy Zobrazit Přejít Záložky Nástroje nápověda

http://linux456.vsb.cz/~nem114/diplomka/spusteni\_ulohy.php

Mozilla Firefox Přehled zpráv

## SPUSTENÍ ULOHY

název: konfigurace OSPF (ukončení do 07.04 14:15)

### Seznam zařízení ulohy

(stisknutím tlačítka se připojí k zařízení)

H1  
RA  
RC  
RD  
SW1  
SW2

```
graph TD
    sRB ---|e0| MHUB
    MHUB ---|10/100fe1| HOST_2
    HD_C ---|e0| RA
    RA ---|e0| SW1
    SW1 ---|10/100fe3| HOST
    SW1 ---|10/100fe1| SW2
    SW2 ---|10/100fe2| HOST_1
```

Hotovo

pavel K3b - Virtuální síťová laboratoř

13:42

Terminal applet form Virtual network laboratory - Mozilla Firefox

```
enable
#config terna
(config)#interface serial 0/0
(config)#ip address 192.168.0.1 255.255.255.0
(config)#no shutdown
```

Terminal applet form Virtual network laboratory - Mozilla Firefox

```
enable
#config terna
(config)#interface serial 0/0
(config)#ip address 192.168.0.1 255.255.255.0
(config)#no shutdown
```

# Java Applet Access Client

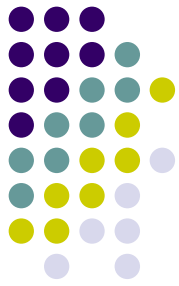


- Launched by access management system in student's browser for each remotely-accessed device in separate window
- Provides access to laboratory device's console and additional functions
  - Allow student alternately connect and disconnect to the console to let multiple students share single device
    - Single student holds the console at each instant
    - student is informed who holds the device at current instant
  - Disconnection warnings before timeslot ending
  - Protects from entering prohibited commands
    - (defined by system administrator)
  - Allows input/output capture to local file system
  - Allows cut&paste insertion of commands into console
  - Informs that tutor started/stopped to access the same device and optionally displays tutor's activity



# Tutor Role

- Tutor can access console of any laboratory device anytime
  - Demonstration mode – student can see what tutor does
  - Hidden mode – student only knows that tutor took over his/her console
    - used for student examination purposes
- Future plan: tutor can passively watch student's activities at any laboratory device



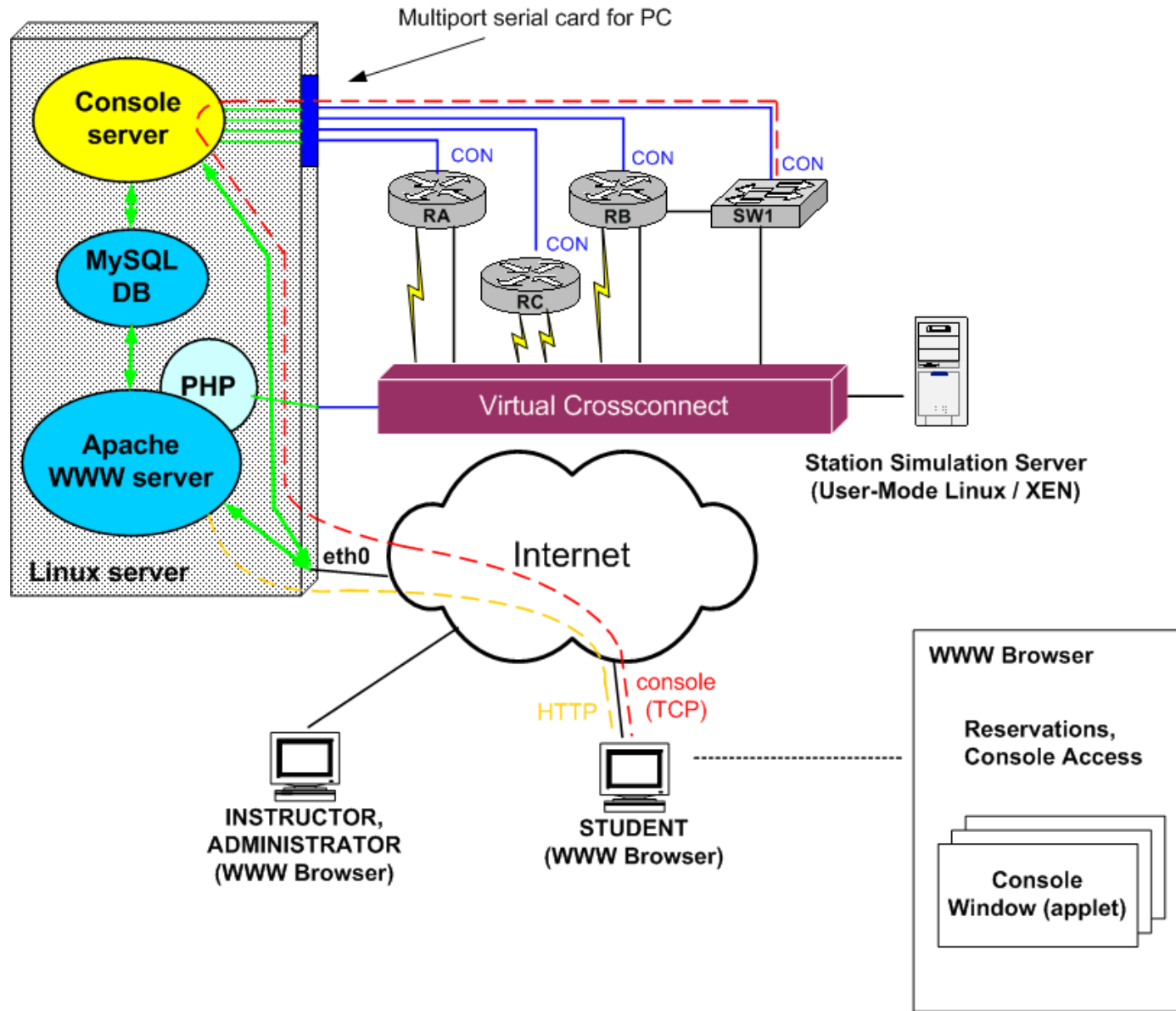
# **Automatic Topology Interconnection System**

# Device Topology Interconnection

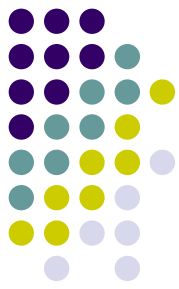


- Before task is made accessible to student(s), required topology has to be interconnected
- We use our own “Virtual Crossconnect”
  - integrates various switching elements
- Topology may be also physically connected by dedicated person informed to do so via email generated by access management system
  - but it is proved unacceptable for real operation

# Position of Virtual Crossconnect in the System Architecture

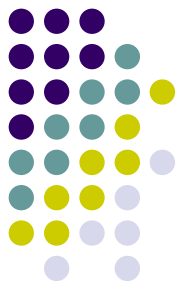


# Why “Virtual” Crossconnect ?



- Implemented using various and multiple physical switching elements
  - ASSSK1, Cisco Catalyst 3500, Catalyst 1900, ...
- Treated like single entity (“virtual crossconnect”) by other parts of system
- All LAN/WAN ports of laboratory devices connected to Virtual Crossconnect ports
- Required topology description is completely independent of types of actually used switching elements

# Virtual Crossconnect Switching technologies



- Serial Ports – our own microprocessor-controlled analog crossbar implementation
  - Current model called ASSSK-1
  - New FPGA-based model under development
- Ethernet Ports
  - VLAN-based interconnection between L3 devices
  - VLAN-tunneling (802.1 QinQ) between switches
    - allows interconnection of trunk links
    - transparent to Spanning Tree and other L2 control protocols



# Currently Used Virtual Crossconnect Switching Elements



**ASSSK-1 (developed by us)**



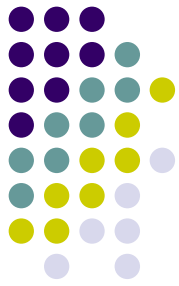
**Cisco Catalyst 3550**  
(supports VLAN tunneling  
and L2 protocol transparency)

**... possibly other switching elements  
in the future**



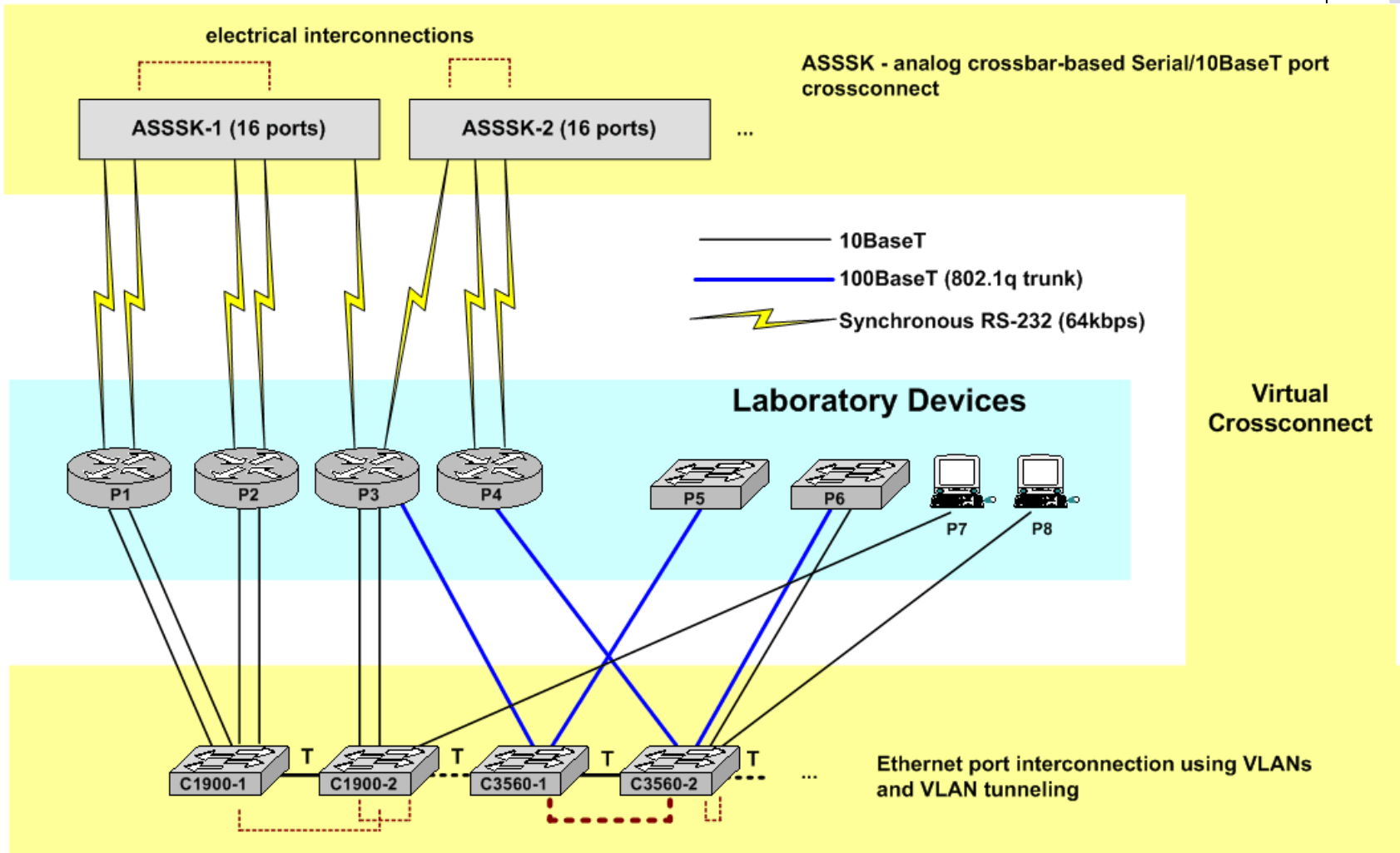
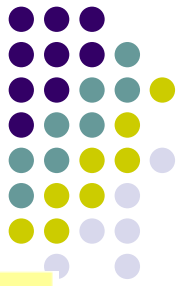
**Cisco Catalyst 1900**  
(cheap solution)

# ASSK-1 Switching Element



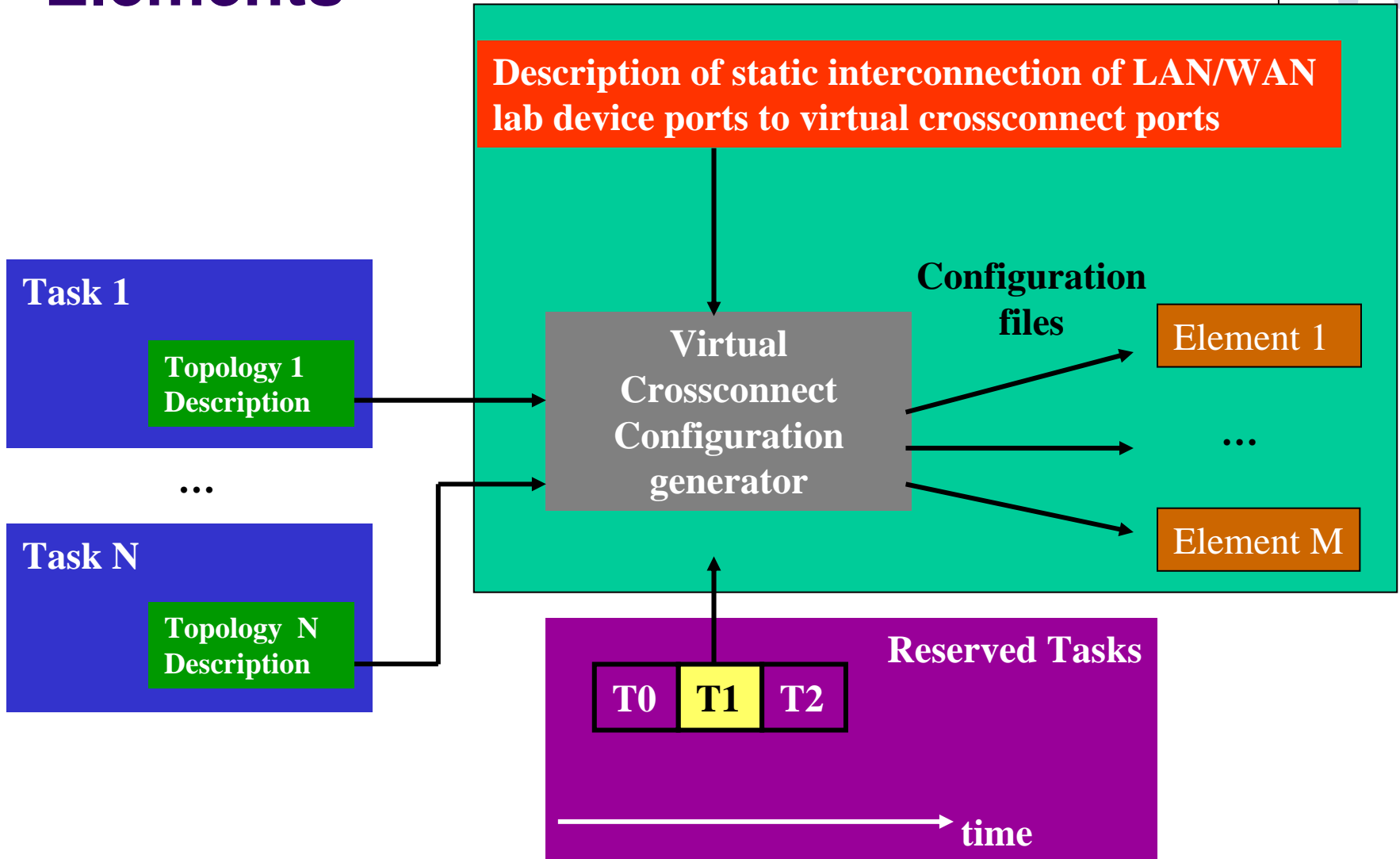
- Modular design
- 16 Serial/Ethernet ports
- Serial ports behave like DCE
- Controlled using IOS-style CLI

# Virtual Crossconnect Components in Action



----- VLAN interconnection      ..... VLAN-tunnel interconnection

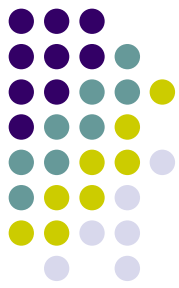
# Generation of Configurations for Virtual Crossconnect Switching Elements



# Virtual Crossconnect - Future plans



- ASSSK-1
  - Various clockrates on serial lines
  - WAN links flapping simulation to let students get experience with real-word WAN troubleshooting
- ASSSK-2 (under construction now)
  - Redesign of ASSSK1 using FPGA
    - smaller, cheaper
  - Only for serial ports inteconnection
    - for Ethernet ports, VLAN-tunneling using standard switches proved more efficient
- ASSSK-3 (under investigation)
  - serial-port crossconnect implementation based on multiport serial card for PC, Linux HDLC/PPP drivers and bridging software
- Enhancement of element configuration generator scripts
  - implement semantic checks of topology definition provided by student who requests his/her own topology



# **Incorporation of Simulated Network Devices**

# What Simulated Devices do we Use ?



- Stations running Linux
  - full user control using text console as with console-controlled network devices
  - fast, efficient, deterministic and reliable networking configuration
- Simulated Cisco 7200-series routers
  - Implemented using open-source DynaMIPS/DynaGEN project
  - Flexible platform to test advanced routing features

# Reasons to Incorporate User Stations into Lab Topologies



- User stations often necessary to test functionality of task solution
- Stations may serve to run various network services and/or practice configuration of these services
  - DHCP, DNS, Syslog, RADIUS, TACACS, ...
- Servers needed to run applications to tests access lists configuration
  - WWW, FTP, Telnet, SSH, ...

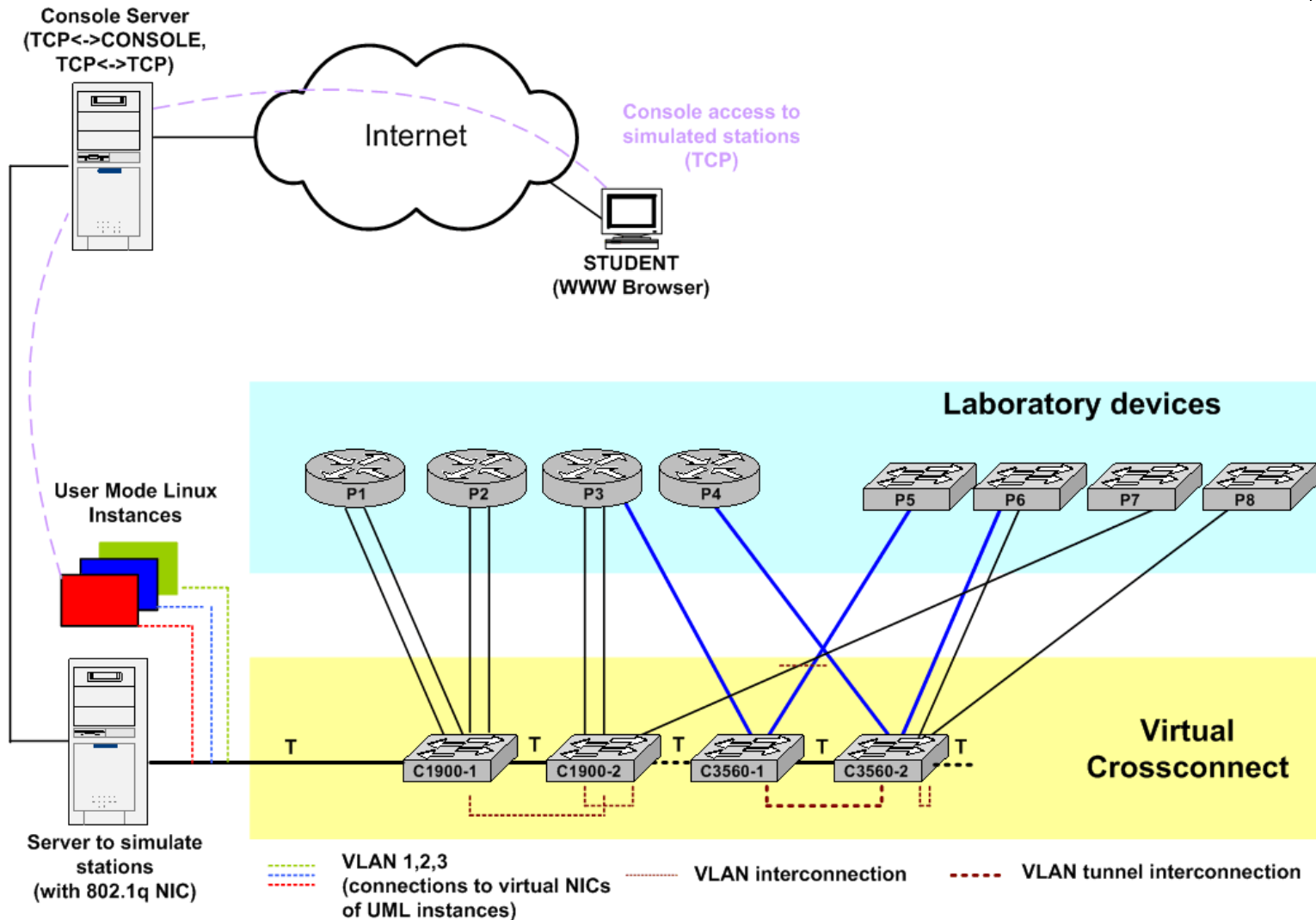
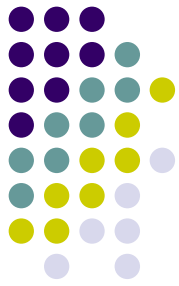


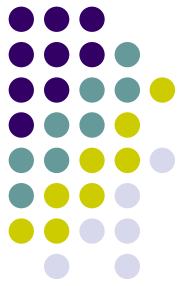
# Advantages of Simulated Devices



- It is ineffective to use multiple physical PCs
  - Physical space requirements, energy consumption, complicated installation/administration
  - No processor or memory intensive application are expected to be ran
- Some virtualization mechanism proved to be useful
  - User-Mode Linux (UML) used currently, usage of XEN considered
- Simulated Ethernet interfaces connected to virtual crossconnect using VLANs
  - Virtual crossconnect architecture was generalized to be able to interconnect devices residing on fixed VLANs with real or other simulated devices

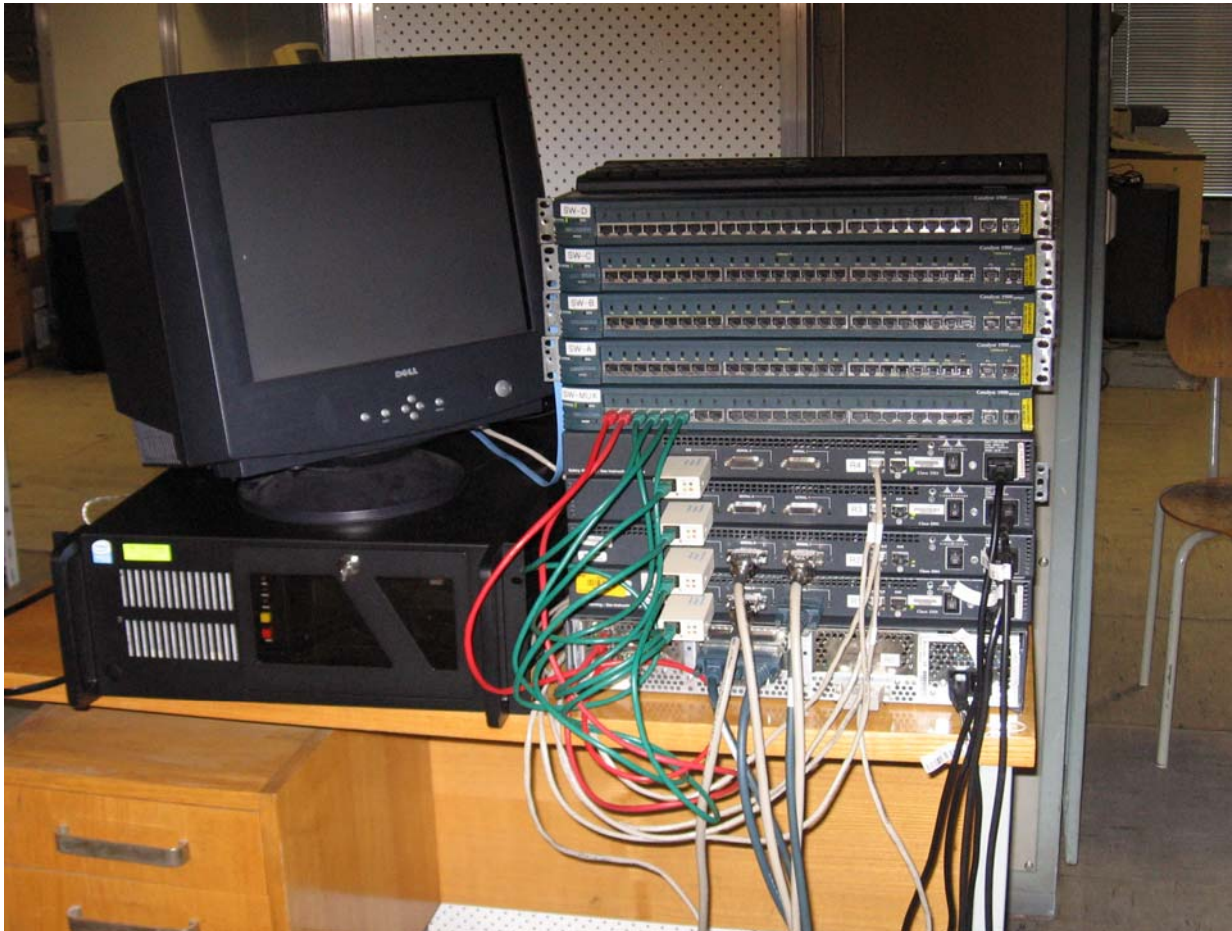
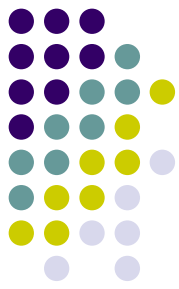
# Integration of Simulated Devices with Virtual Crossconnect





# Today's Functional Single-Site Implementation

# The Very First Virlab Implementation

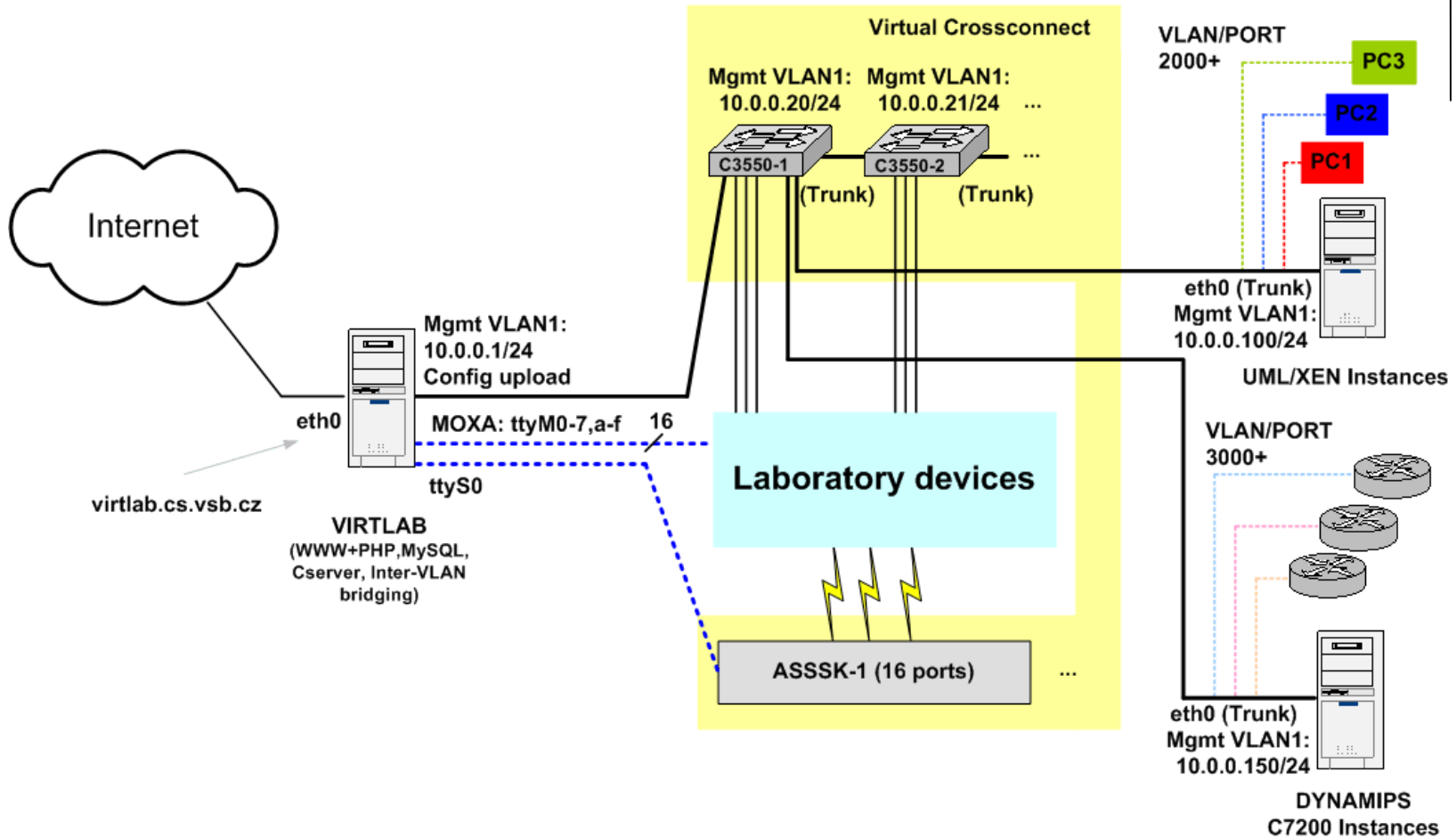


# Today's Virlab

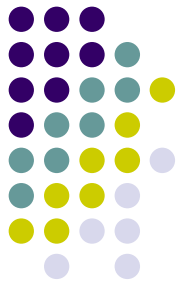


- Two 45U racks incorporating
  - Access Management System Server
    - (Virtlab Server)
  - Virtual Crossconnect
  - UML server
  - DynaMIPS server
  - C2500/2600/4000 routers
  - C1900 switches
  - C2500 switches
  - C5500 switch with RSM

# Today's Virlab Internal Architecture



# System Security



- Web interface – uses HTTPS
  - users authenticated using passwords stored in database or through LDAP
- Console access
  - One-time password authentication
  - Digitally-signed applet
- Access via firewalls taken into account
  - single fixed TCP used for console access
- Extensive logging of user's activity
  - Access reservation system logins
  - Console access



# Architecture Extension

## Dynamic Mapping of Network Devices used for Particular Reserved Tasks

(Current Work)



# The General Idea (1)



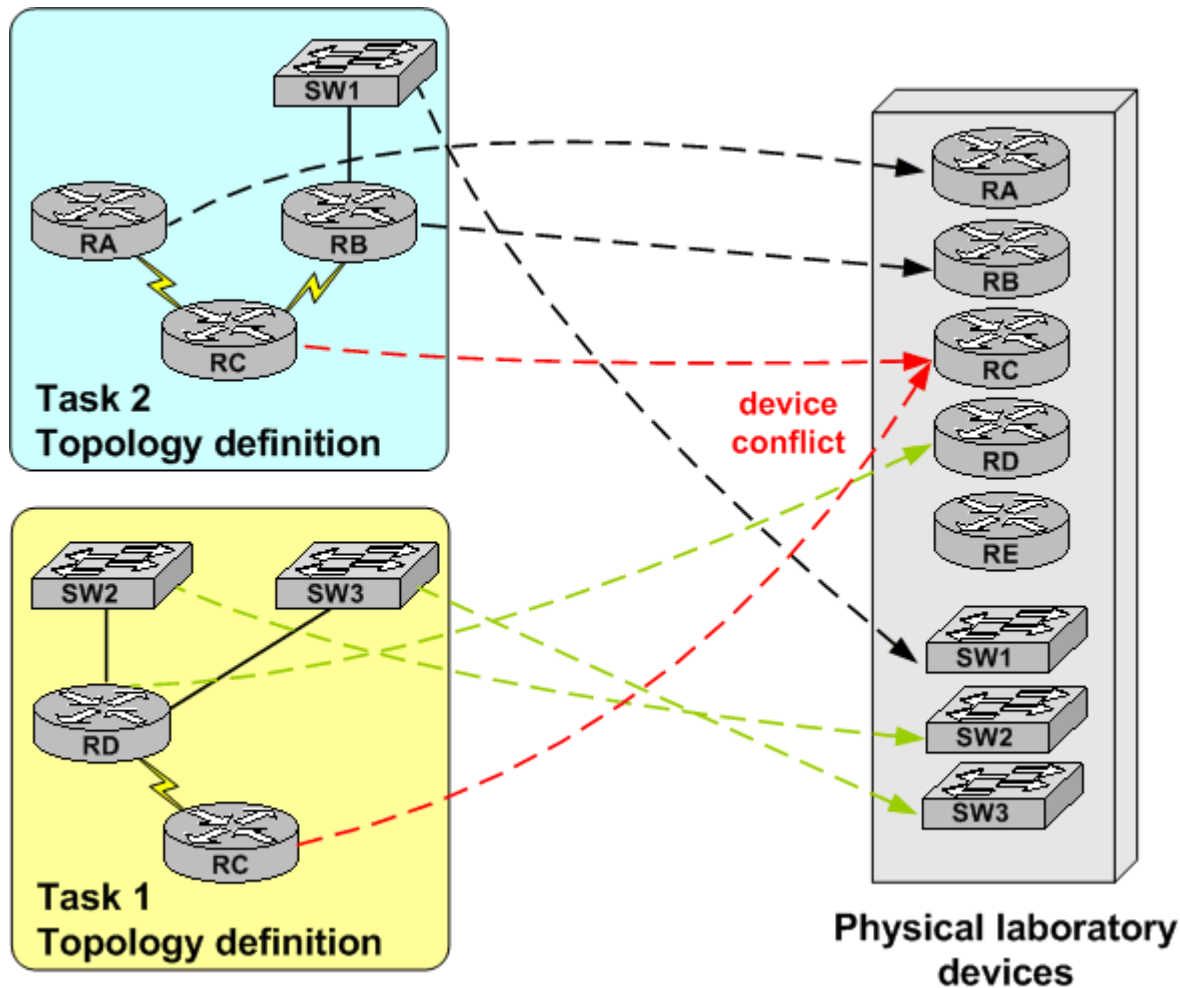
- Fixed-length and fixed-beginnings timeslots proved inefficient
  - Students are not interested in some scheduled tasks but there is a contention for timeslots with other more interesting tasks
- Since we are able to interconnect topologies automatically, it is better to let students choose ANY task at ANY timeslot
  - fixed timeslots abandoned, student may reserve any time interval up to his/her weekly quota
- Task scheduler role no longer needed
  - notice-board used only to remember what task was reserved for what time interval

# The General Idea (2)

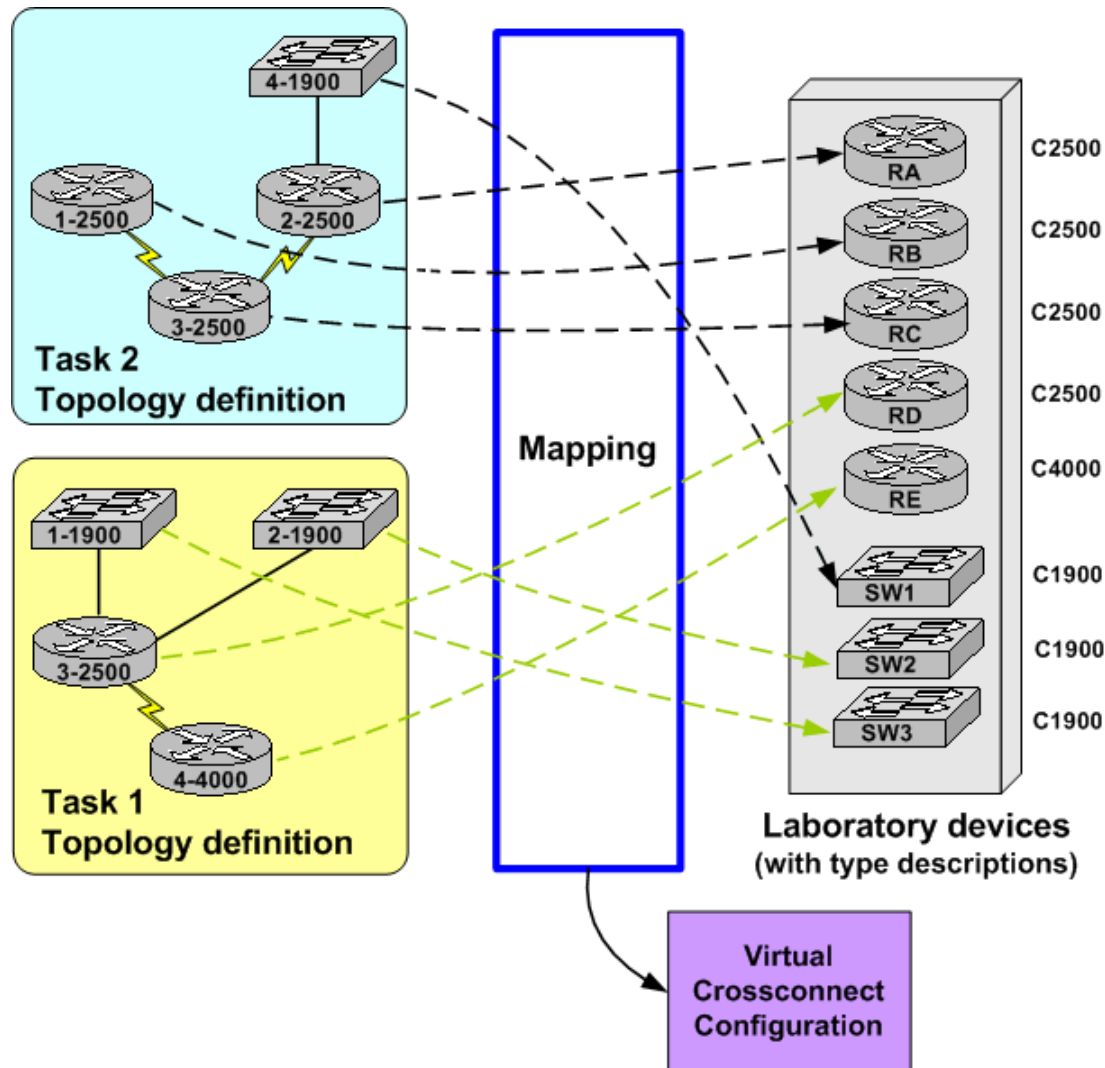
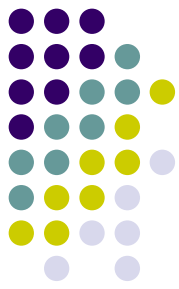


- To utilize virtual laboratory equipment efficiently, it is useful to let multiple tasks be reserved in parallel if there is enough network devices
- To let multiple tasks be reserved in parallel without unnecessary limitations and device conflicts, it is needed to decouple task definitions from physical device identities
- Physical devices used for task are chosen dynamically with respect to other devices used at the same time

# Problem of Task Definition Coupled with Physical Device Identities



# Dynamic Mapping of Task to Physical Devices During Task Reservation Procedure



# Function of Dynamic Device Mapping



- Mapping takes place when student reserves task for particular time interval for himself/herself
- Network device types, numbers of interfaces and other features have to be taken into account
- Multiple tasks may be mapped in parallel
  - feasibility determined during mapping process at reservation time

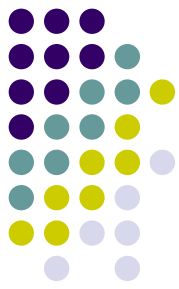


# **Distributed Multiple-Site Architecture**

**(Current Work)**

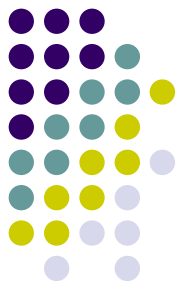
# Multiple-Site Architecture

## Basic Paradigm

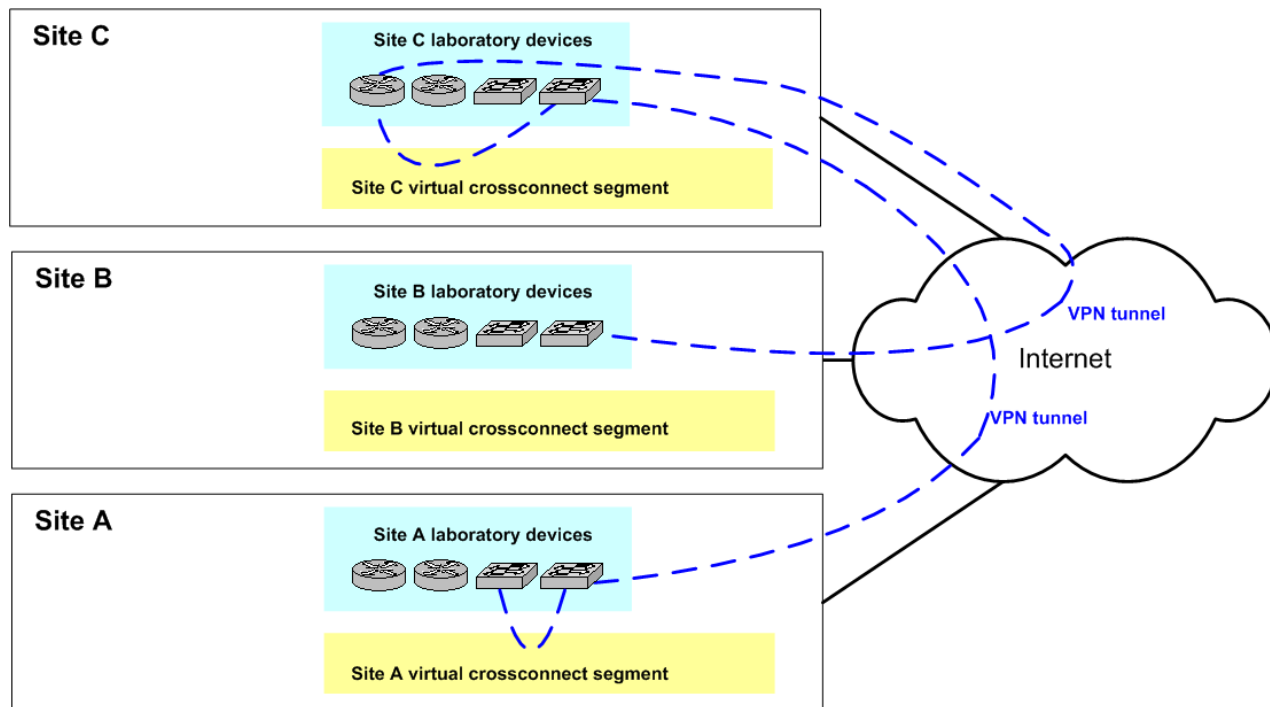


- Multiple cooperating sites may share laboratory devices transparently
  - Constraints to offer site's lab devices for other sites may be specified
  - Local mappings are preferred to limit inter-site traffic
- Fully-decentralized architecture allows independent operation of individual sites if other sites become unavailable
- Distributed nature is hidden to student
  - he/she accesses device consoles the same way regardless of physical target device placement
  - virtual topology between devices of multiple sites behaves the same way as single-site topology

# Distributed topologies

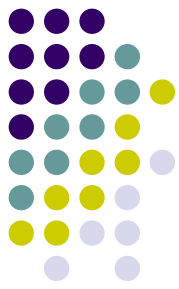


- Distributed virtual topologies may be constructed using Internet tunnels
  - Tunneling of Ethernet links is implemented now
  - We are also working on HW/SW solution to tunnel serial (WAN) links



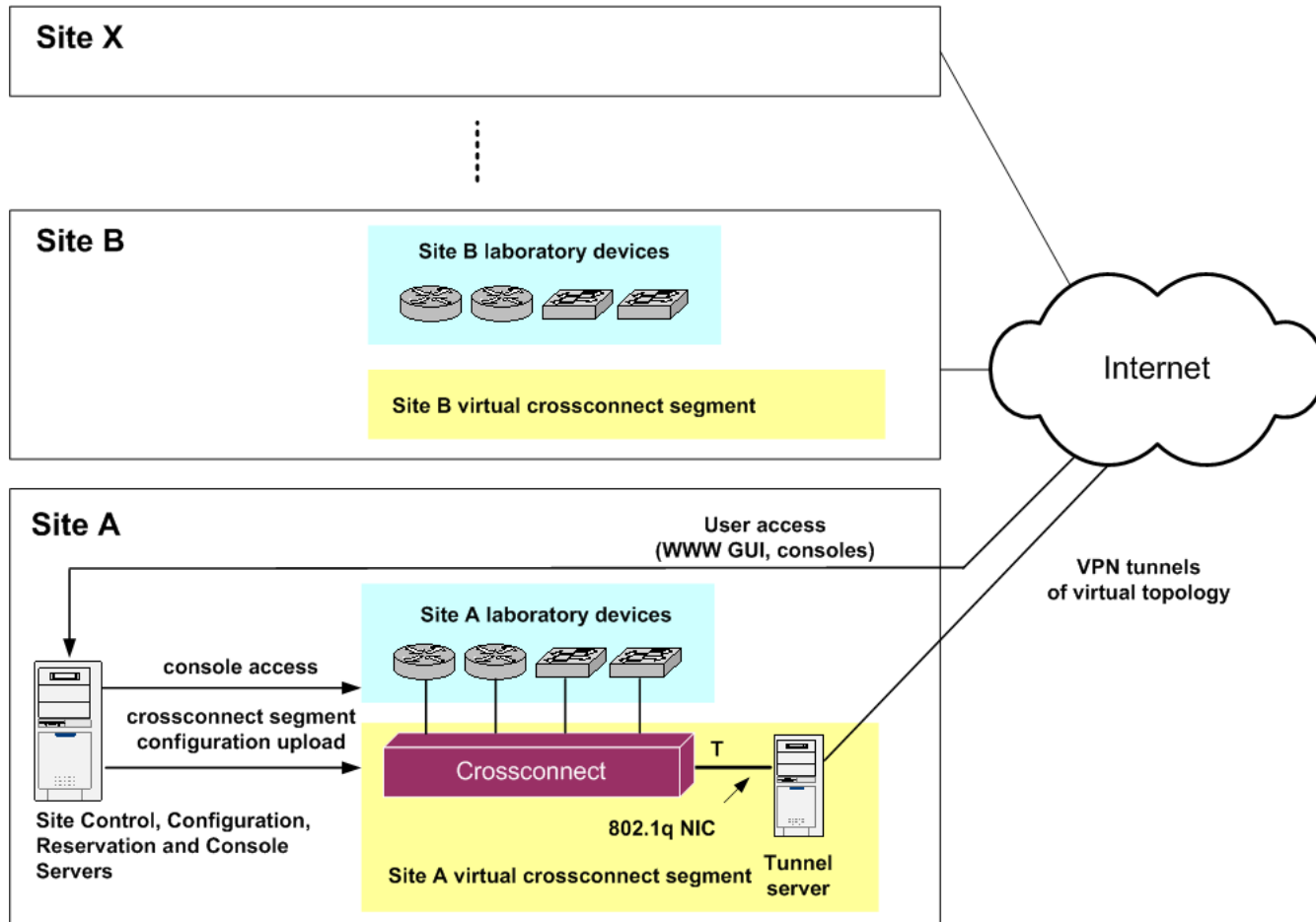
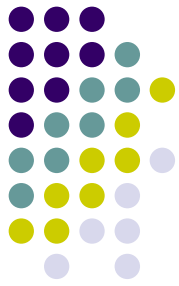


# Implementation of Distributed Topologies

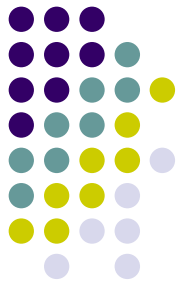


- Virtual Crossconnect concept was generalized to support multiple-site topologies
- Tunnel Servers allow to extend VLAN-based crossconnections over Internet tunnels between sites (802.1q in UDP)
- Virtual Crossconnect is now treated as set of Virtual Crossconnect Segments in individual sites which form single Distributed Virtual Crossconnect
- Layer 2 frames tunneling allows transparent operation of multiple layer 3 protocols

# Multiple-Site Architecture Basic Components

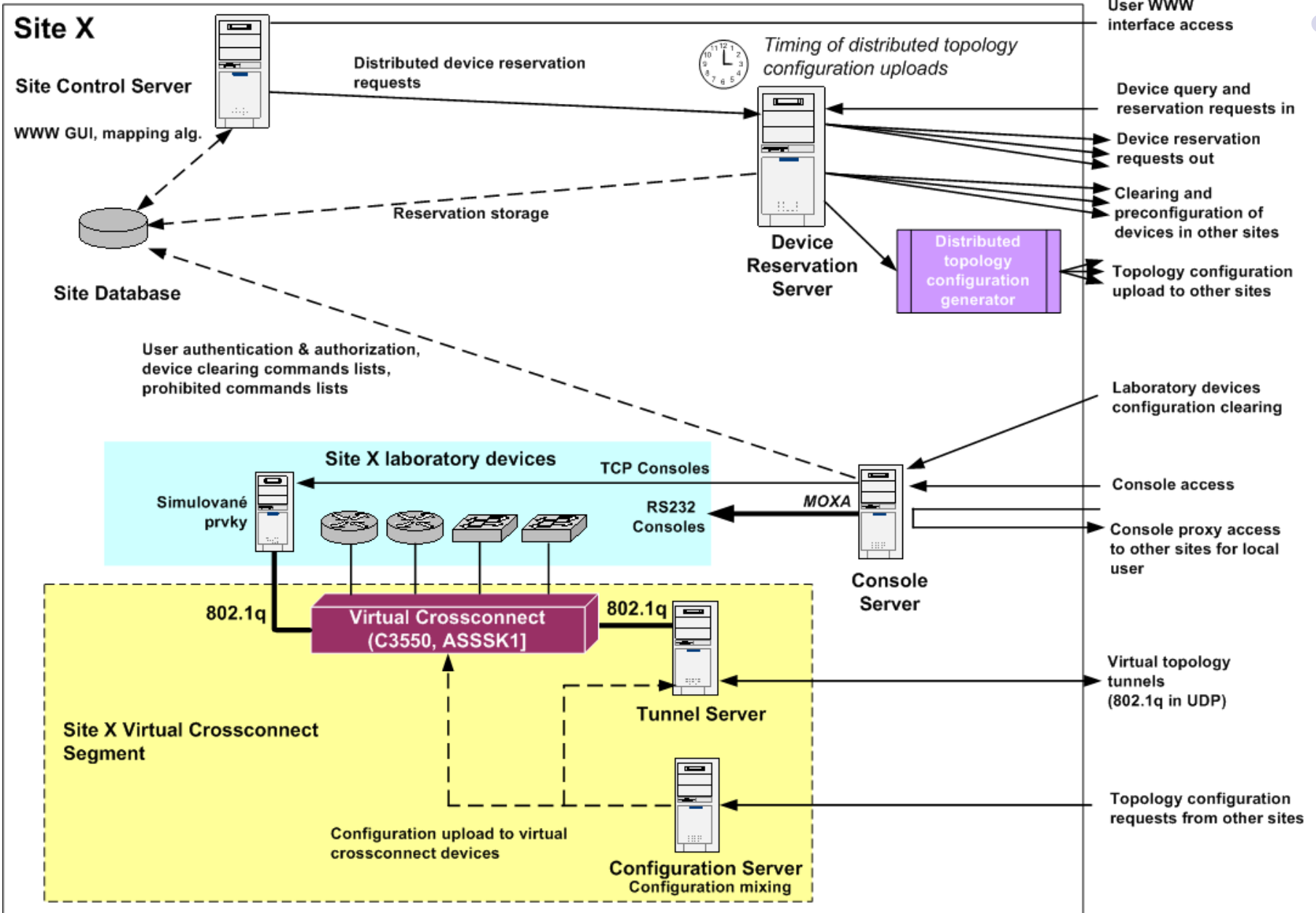


# Function of Control Components



- **Virtlab Server**
  - Provides access management system GUI, console access GUI and device mapping algorithm
- **Reservation Server**
  - Keeps track of site devices' reservations
- **Configuration Server**
  - accepts distributed virtual crossconnect configuration requests
- **Console Server**
  - allows access to consoles of sites' devices
- **Tunnel Server**
  - Tunnels traffic between devices in different sites

# Interaction of System Components



# Advantages of Multiple-Site Architecture



Multiple cooperating teaching/research institutions are able to

- implement large WAN topologies using devices of multiple participant sites
  - behavior of real-world topologies may be studied
- specialize to buy expensive special devices and share them with others