The DRAGON Project

Dynamic Resource Allocation over GMPLS Optical Networks

Presented to JET OTN Workshop

April 19, 2005

Washington, DC

National Science Foundation

- **Jerry Sobieski**

  Mid-Atlantic Crossroads (MAX)
- **Tom Lehman**

  University of Southern California

  Information Sciences Institute (USC ISI)
- **Bijan Jabbari**

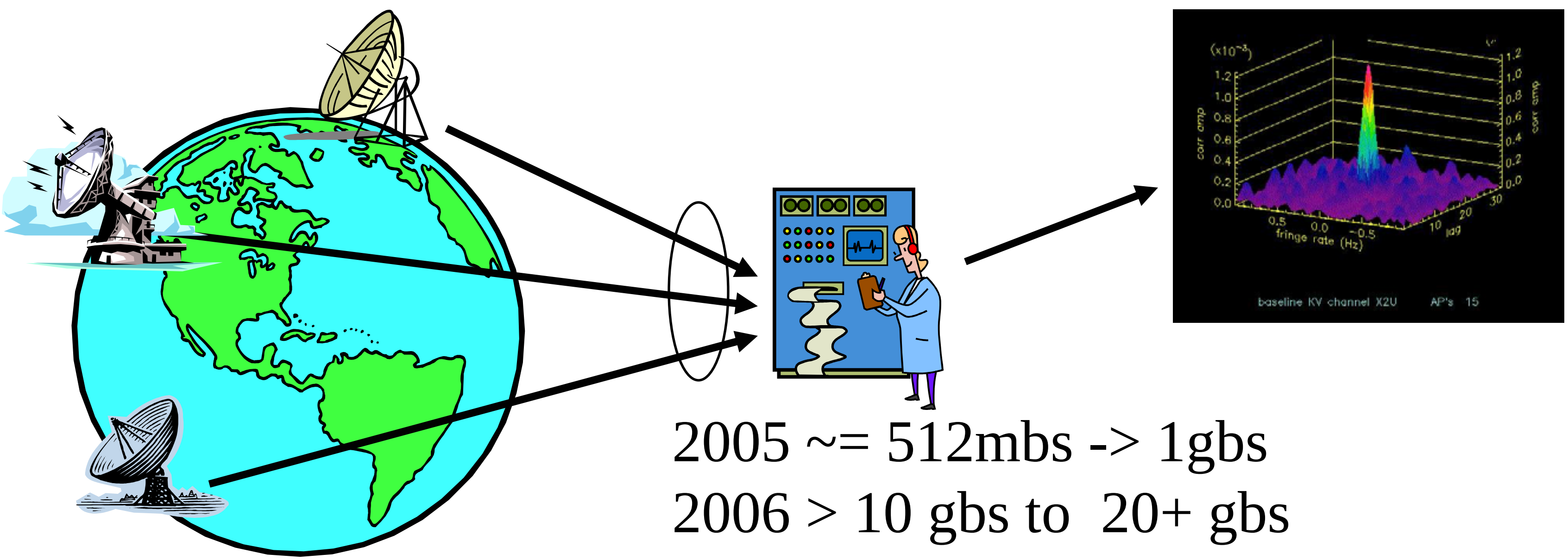
  George Mason University (GMU)
- **Don Riley**

  University of Maryland (UMD)

- **A small but very challenging set of "e-Science" applications are emerging:**
  - Globally distributed science, science teams, and instruments,
  - Network intensive:
    - Large capacity flows
    - Real time and near real time data transfer
    - Scheduled science resources (e.g. access to resources)
- **Existing networks find it very difficult to meet the performance requirements of these applications…**

E-science Example: E-VLBI

- **Electronic Very Long Baseline Interferometry (eVLBI)**



2005 ~= 512mbs -> 1gbs
2006 > 10 gbs to 20+ gbs
2008 > 20 gbs to 40+ gbs

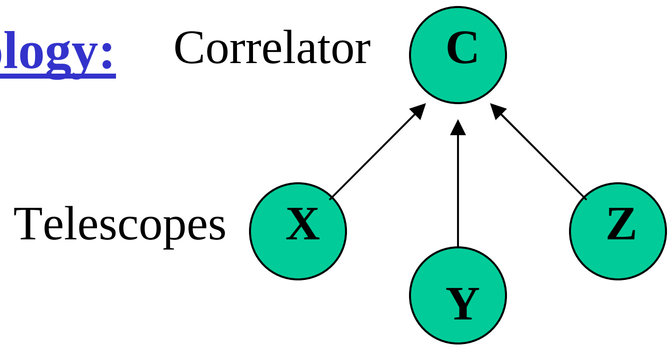
2004 = 128 mbs
2006 > 2 gbs
2008 > 4+ gbs

E-VLBI Requirements

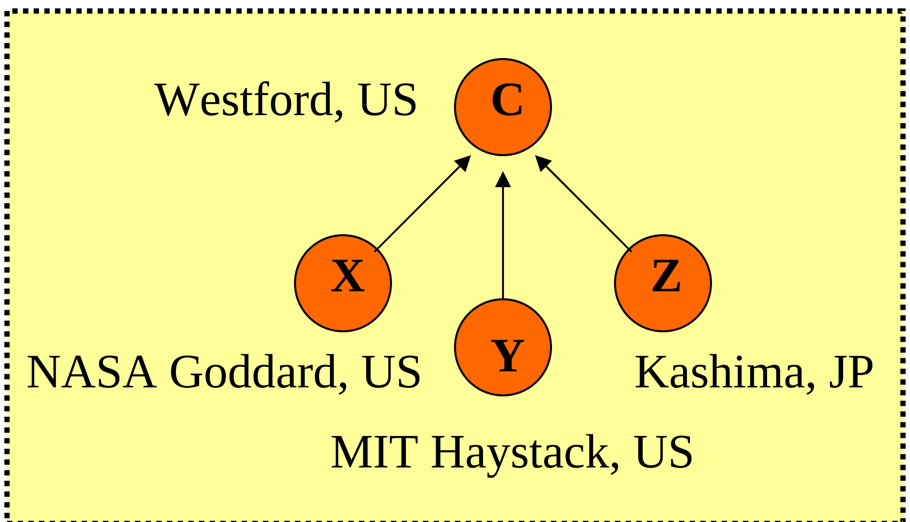
- **From 2 to as many as 20+ radio telescopes distributed around the world**
  - Each generating 500mbs real time streams (this year)
  - Converging on a single computational cluster
- **Realtime correlation requires network resources that are:**
  - Deterministic in terms of loss, latency, and jitter
  - Repeatable
  - Scheduleable – must be coordinated with other resources such as the availability of the telescope itself.
  - Rapidly provisioned, under one minute to establish the topology
- **The network is an integral part of the application itself –**
  - An application specific topology (AST) must be instantiated “en masse” to run the application – I.e. all network resources must be provisioned as a whole
  - This physical AST will vary with the location of available/required nodal resources

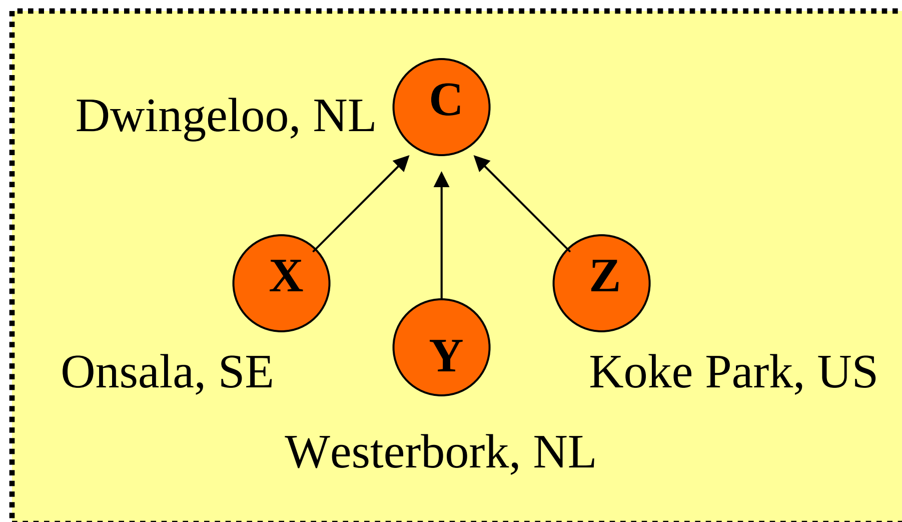
The “Application Specific Topology”

Logical e-VLBI Topology:



Physical Instantiations of the Application Specific Topology





What are the generic network requirements of these Super-Apps?

- **Dedicated Network Resources**
  - These applications want/need their own network resources, They do not care to “play fair” with other traffic.
- **Deterministic Network Performance**
  - Network performance must be consistent, predictable, and repeatable
- **Reservable and Schedulable**
  - The network must insure that the resources will be available when needed, and for as long as needed.
- **Very High Performance**
  - These applications require resources that often exceed current IP backbones.
- **Dynamically Provisioned**
  - The topologies, the performance requirements, the priorities, and purpose of the applications are not static and will vary from one run to the next.
- **Application Specific Topologies**
  - All resources must be allocated and provisioned as a whole.

What is the DRAGON Project?

- **Dynamic Resource Allocation over GMPLS Optical Networks**
  - DRAGON is a four year project funded by the US National Science Foundation (NSF)
  - Testbed deployed in the Washington DC metro area
- **Purpose:**
  - To develop/integrate network hardware and software technologies that can support dynamic, deterministic "light path" services.
  - To demonstrate these "light paths" services with real applications and over real network(s)

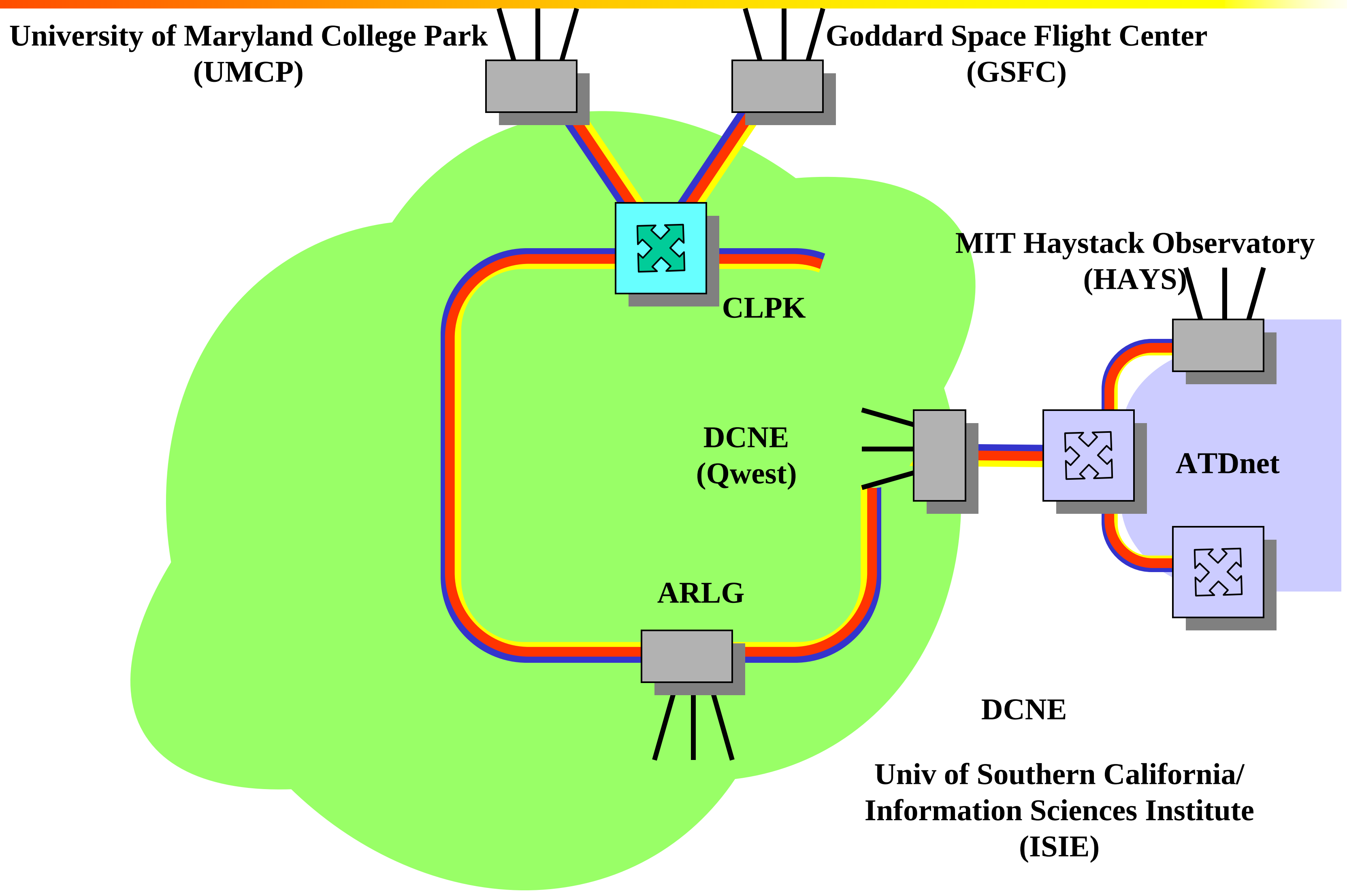
DRAGON Participants

- Mid-Atlantic Crossroads (MAX)
- USC/ Information Sciences Institute (ISI-East)
- George Mason University (GMU)
- MIT Haystack Observatory
- NASA Goddard Space Flight Center (GSFC)
- University of Maryland (UMCP)
- Movaz Networks (commercial partner)
- NCSA ACCESS
- US Naval Observatory

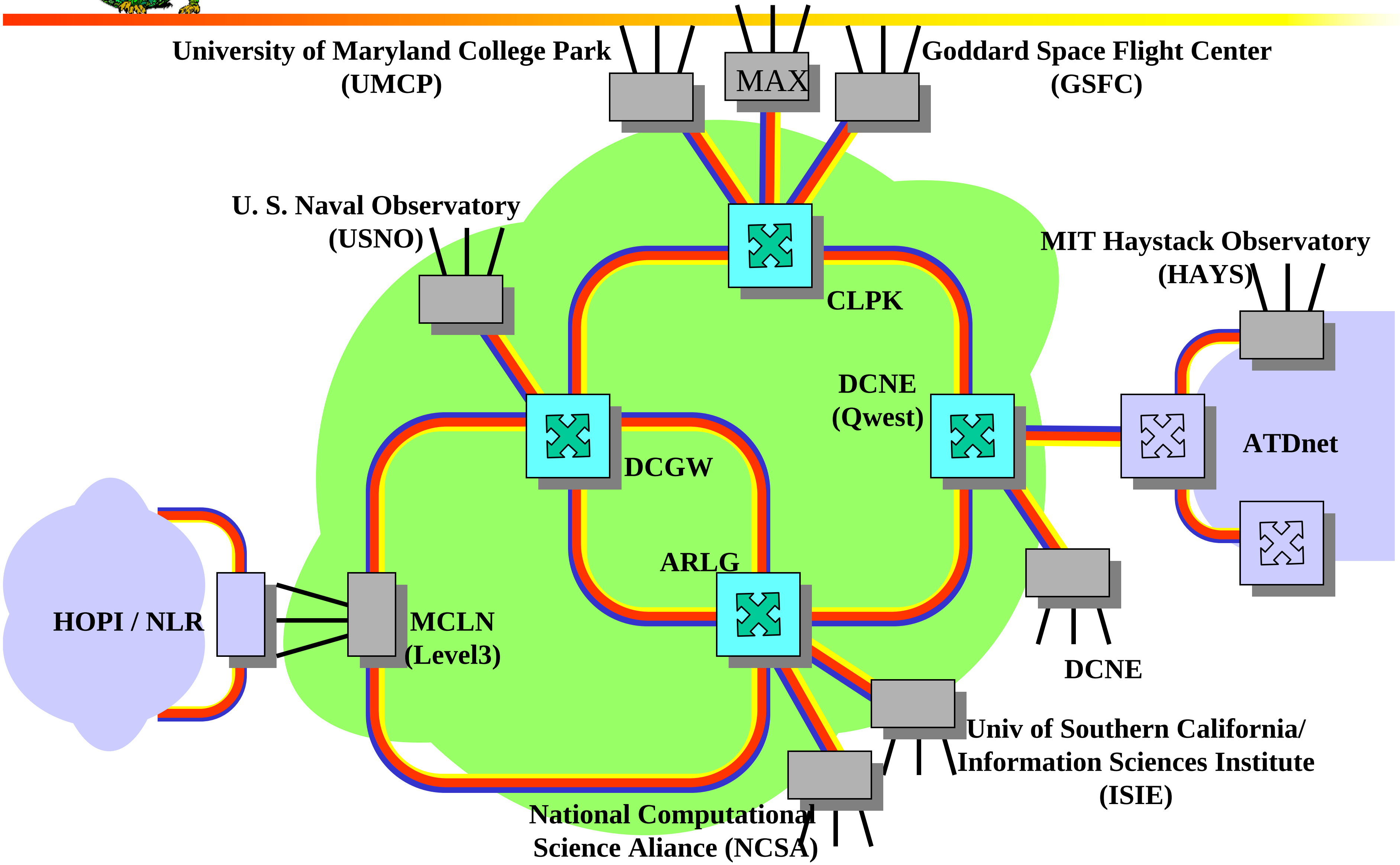
Project Features and Objectives

- **All-Optical metro area network**
  - Reduce/eliminate OEO in the core,
  - Allow alien waves to penetrate demarcs – safely, dependably.
- **GMPLS protocols for dynamic provisioning**
  - Addition of CSPF Path Computation algorithms for wavelength routing
- **Inter-domain service routing techniques**
  - Network Aware Resource Broker (NARB) for service advertising, inter-domain ERO generation, AAA
- **Application Specific Topology Description Language**
  - Formalized means to describe the application topology and network service requirements
- **Integration with real applications:**
  - E-VLBI
  - HD-CVAN

The DRAGON Testbed
Washington, D.C. Metropolitan Area



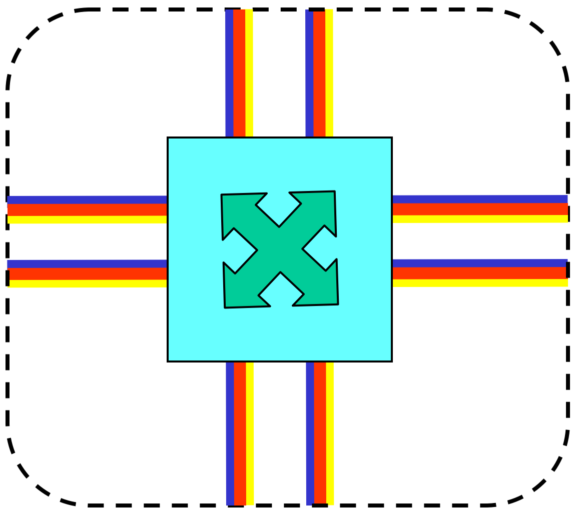
The DRAGON Testbed
Washington, D.C. Metropolitan Area

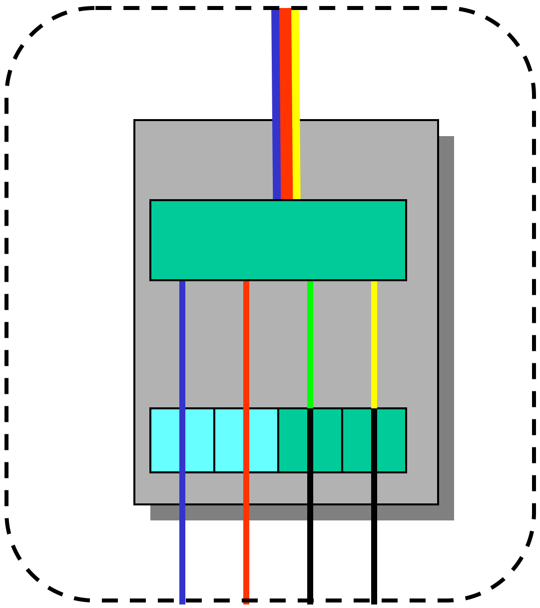


DRAGON Photonic Architecture

- **Principles:**
  - Standard practice of OEO engineering at every node is no longer necessary in metro/regional scale networks
    - Waves can be progressed without regen in most cases
    - Allow the user/client to define the transport
  - Core switching nodes should be optically transparent:
    - Any wavelength from any port to any other port
    - Encoding and Framing agnostic
  - Regeneration provisioned *only* as a service function at core nodes:
    - To provide wavelength translation to avoid blocking conditions
    - To provide signal integrity iff network diameter and service specs require it
  - OEO transponders are used at edge only for ITU translation
    - External ITU wavelength signaling and sourcing is encouraged
  - All wave paths are dynamically allocated using GMPLS protocols
    - Extensions for CSPF path computation and inter-domain are new

DRAGON Generic Architectural Cells





Core Wavelength Switching Primitive Cell

- -All waves are C-Band ITU compliant on 100 Ghz ITU spacing
- -Any wave can be individually switched from any input port to any output port
- -Each port goes to either a) another core switching cell, or b) an edge cell
- -Other wavelengths outside the C-Band are extinguished on entry and are not progressed thru the switch.
- -The switching cell can block any/all input waves on any input port
- -The switch is not sensitive to the content, framing of any data plane wave.

Edge Service Introduction and Validation Cell

- -Client interfaces provide wavelength conversion to ITU grid lambdas
- -External wavelength interfaces verify conformance of customer provisioned waves to network constraints
- -Can also be used at core nodes to provide wavelength translation

An Optical Architecture

- **DRAGON is developing a Dynamic Optical Architecture paper**
  - Consolidate the lessons learned and other best common practices we use to build the DRAGON core network
  - Incorporates a number of issues – not just optical layer engineering, but how we need to integrate the various layers into a common service architecture

Commercial Partner: Movaz Networks

- **Private sector partner for the DRAGON project**
  - Provide state of the art optical transport and switching technology
  - Major participant in IETF standards process
  - Software development group located in McLean Va (i.e. within MAX)
  - Demonstrated GMPLS conformance

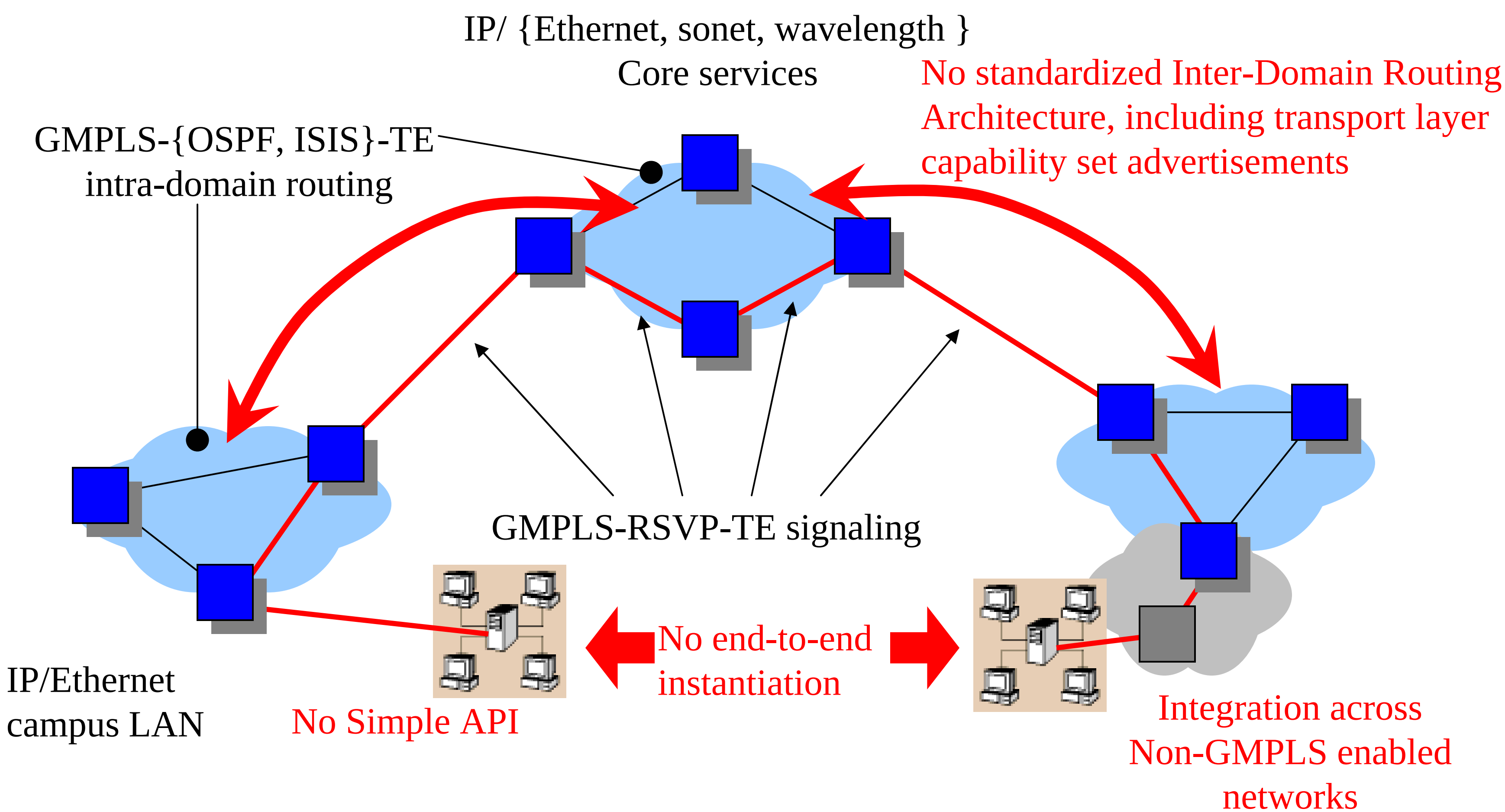
- MEMS-based switching fabric
- 10 ports X 40 waves/port
- 9.23"x7.47"x3.28" in size
- Integrated multiplexing and demultiplexing, eliminating the cost and challenge of complex fiber management

Movaz iWSS prototype switch installed at the University of Maryland

Movaz Partnership: New Technology

- **Movaz and DRAGON will be deploying early versions of new technology such as:**
  - Reconfigurable OADMs
  - Alien wavelength conditioning
  - Tunable wavelength transponders
  - Possibly other digital encoding formats such as RZ, DPSK, etc.
- **The development and deployment plans of selected technologies are part of the annual review cycle**

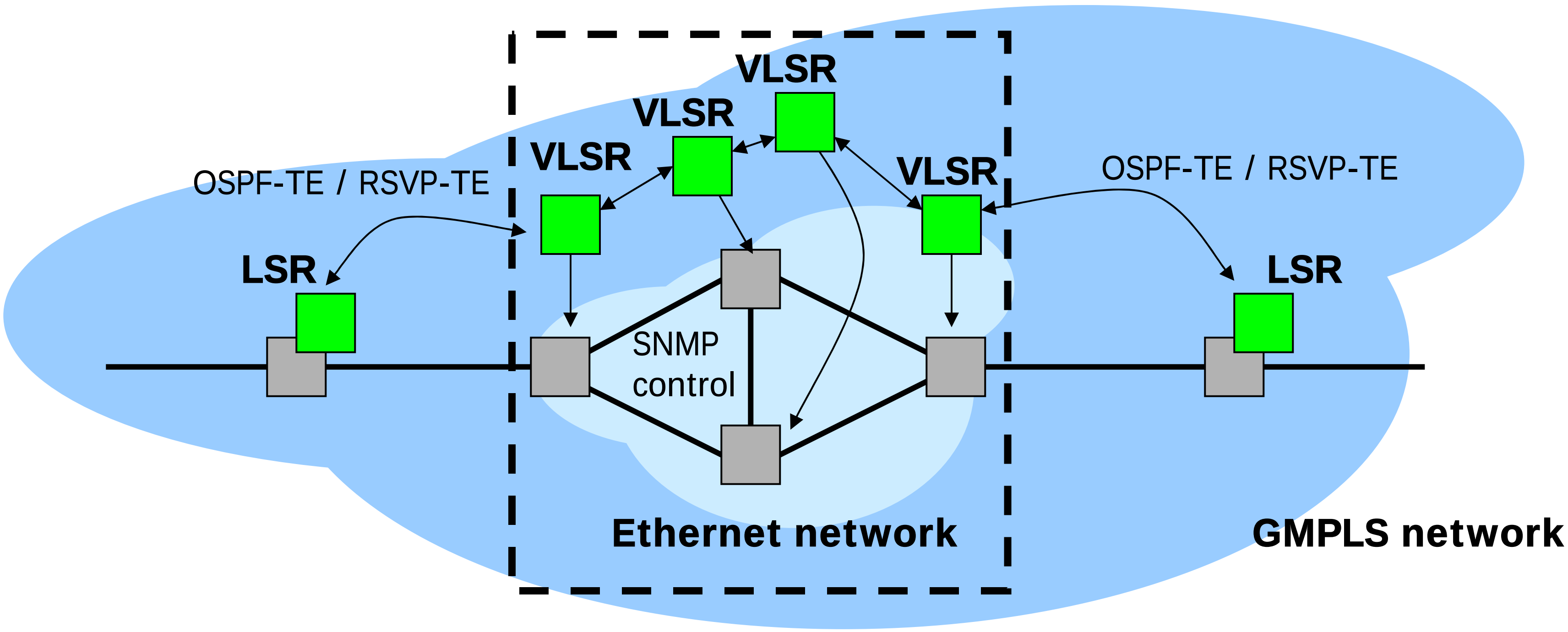
End to End GMPLS Transport
What is missing?



Virtual Label Switched Router: VLSR

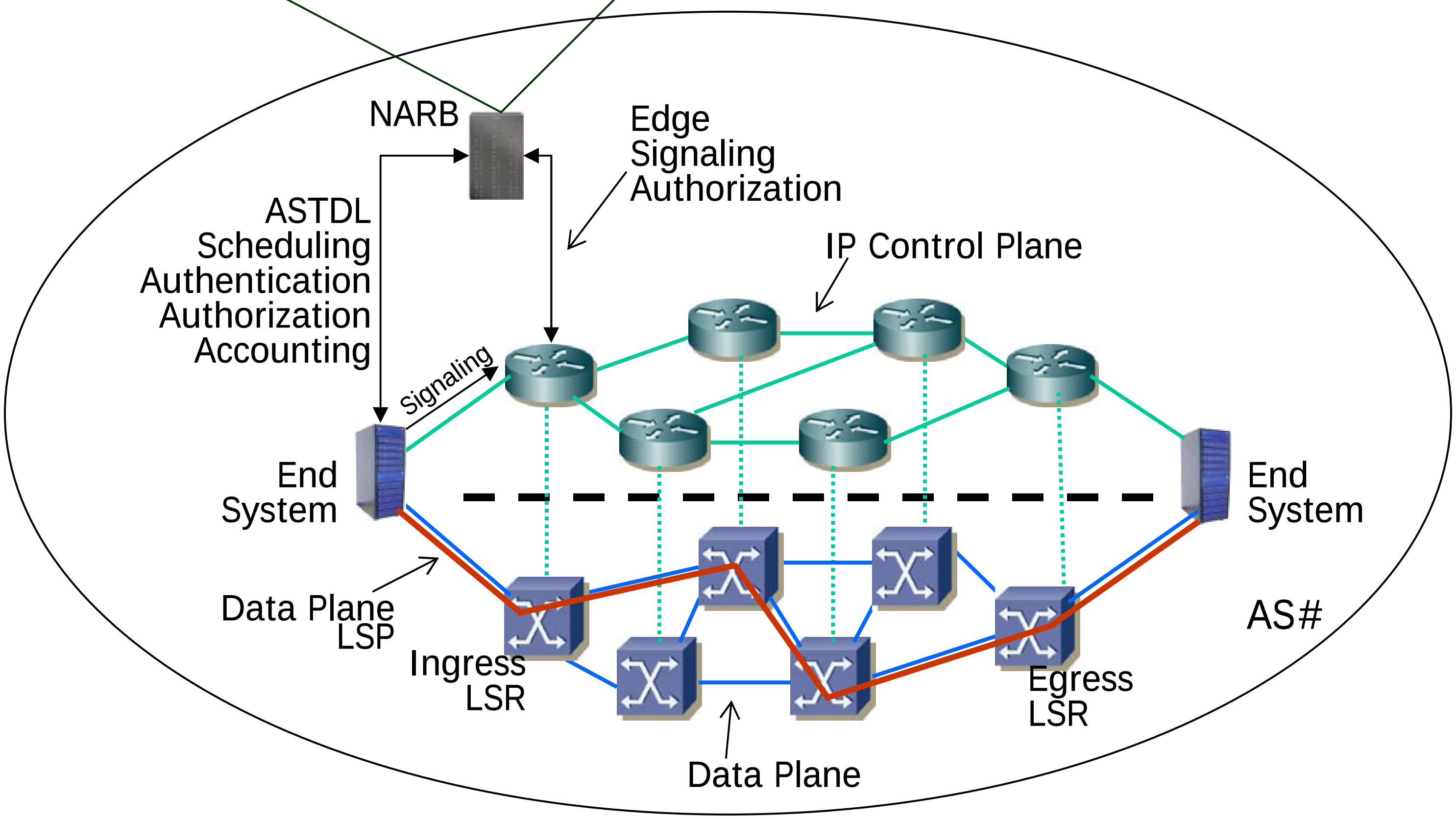
- **Many networks consist of switching components that do not speak GMPLS, e.g. current ethernet switches, fiber switches, etc**
- **Contiguous sets of such components can be abstracted into a Virtual Label Switched Router**
- **The VLSR implements Open Source versions of GMPLS-OSPF-TE and GMPLS-RSVP-TE and runs on a Unix based PC/workstation**
  - Zebra OSPF extended to GMPLS
  - KOM-RSVP likewise
- **The VLSR translates GMPLS protocol events into generic pseudo-commands for the covered switches.**
  - The pseudo commands are tailored to each specific vendor/architecture using SNMP, TL1, CLI, or a similar protocol.
- **The VLSR can abstract and present a non-trivial internal topology as a “black box” to an external peering entity.**

VLSR Abstraction



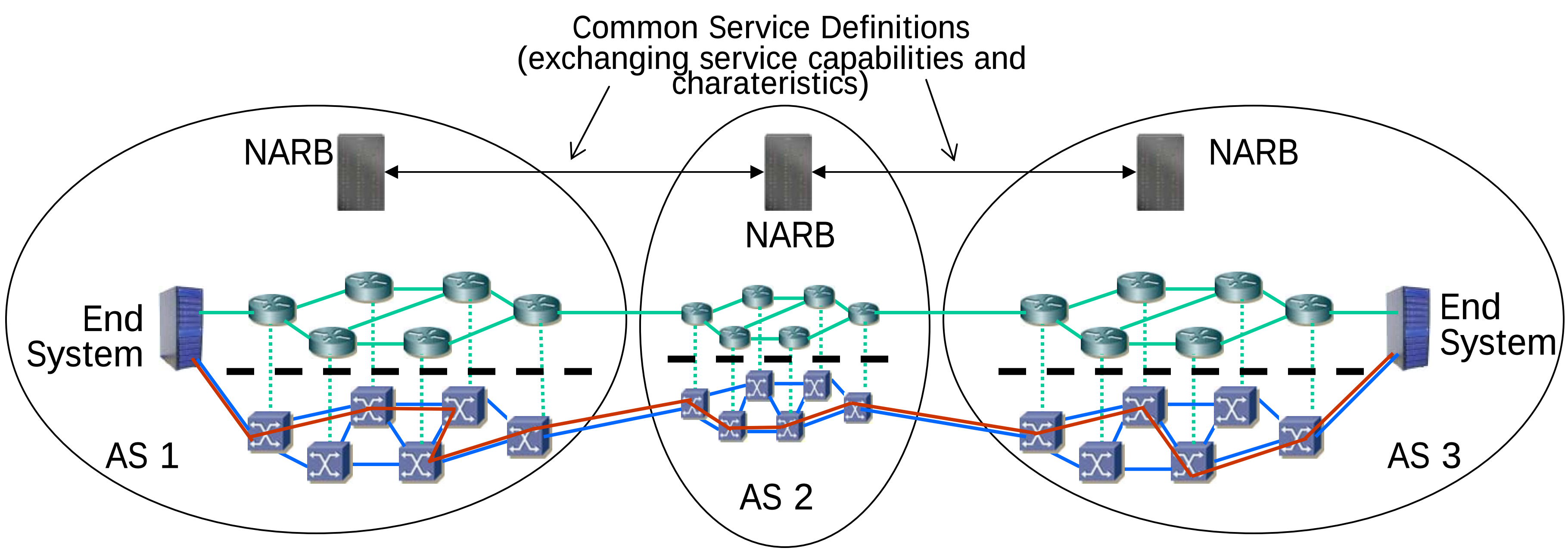
Network Aware Resource Broker (NARB) Functions – IntraDomain

- IGP Listener
- Path Computation
- Scheduling
- Edge Signaling Authentication
- Edge Signaling Enforcement
- ASTDL Induced Topology Computations
- Authorization (flexible policy based)
- Authentication
- Accounting



Network Aware Resource Broker (NARB) Functions - InterDomain

- **InterDomain NARB must do all IntraDomain functions plus:**
  - EGP Listener
  - Exchange of InterDomain service definition(s)
  - InterDomain path calculation
  - InterDomain AAA policy/capability/data exchange and execution



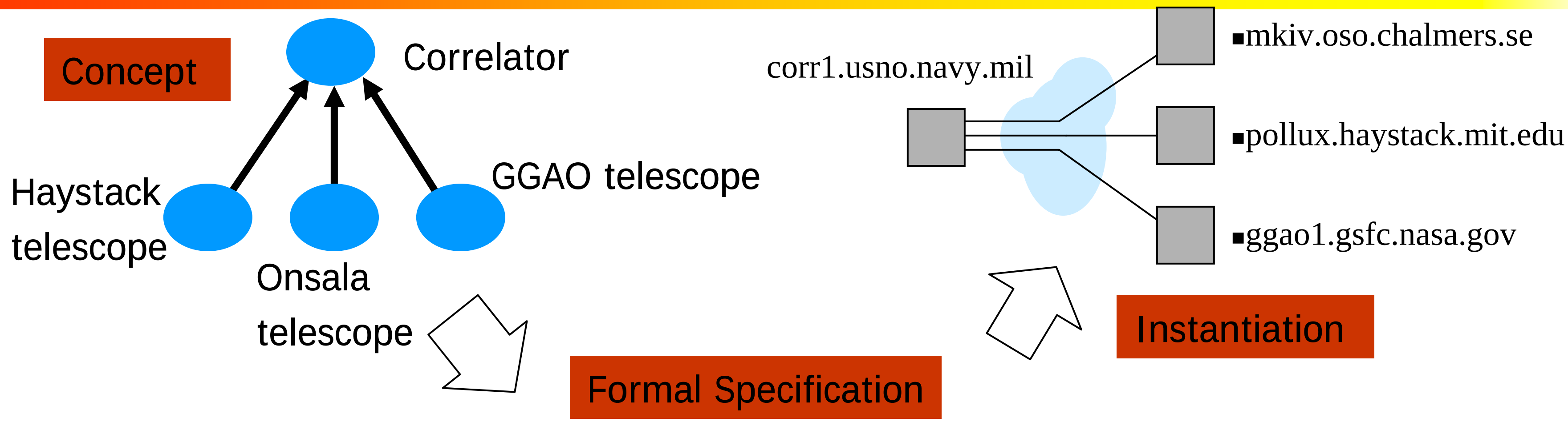
The Common Service Definition

- **One of the painful facts we have discovered as we study and try to address inter-Domain service routing has been the in-ability to establish end-to-end light paths with a clear and consistent "performance" metric:**
  - i.e. How do we verify that the light path requested was actually established?
  - What were the requested service characteristics associated with a specific request?
  - Which domains along the path can actually meet those service characteristics?
- **DRAGON has described a "Common Service Definition" model for the creation of services that can be provably consistent, predictable, and verifiable end to end.**
  - **http://dragon.maxgigapop.net/public/Common-Services-Definition-v0.6.pdf**
  - The document was the product of much discussion within the JET community, the GLIF community, and particularly the ONT projects themselves (CHEETAH, OMNInet, DRAGON, and USN).
  - Comments are welcomed and encouraged; The document is meant to stir discussion and spur further development of the concept.

Application Specific Topology Description Language: ASTDL

- **ASTDL is a formalized definition language that describes complex topologies**
  - By formally defining the application's network requirements, service validation and performance verification can be performed ("wizard gap" issues)
  - Formal definition allows advanced scheduling – which must still be integrated with non-network resources such as computational clusters, instruments, sensor nets, storage arrays, visualization theatres…
- **ASTDL includes run time libraries to instantiate the topology and link in the other resources of the application**
  - Application topologies consist of multiple LSPs that must be instantiated as a set.
  - Resource availability must be dependable and predictable, i.e. resources must be reservable in advance for utilization at some later time

Application Specific Topology Description Language - ASTDL

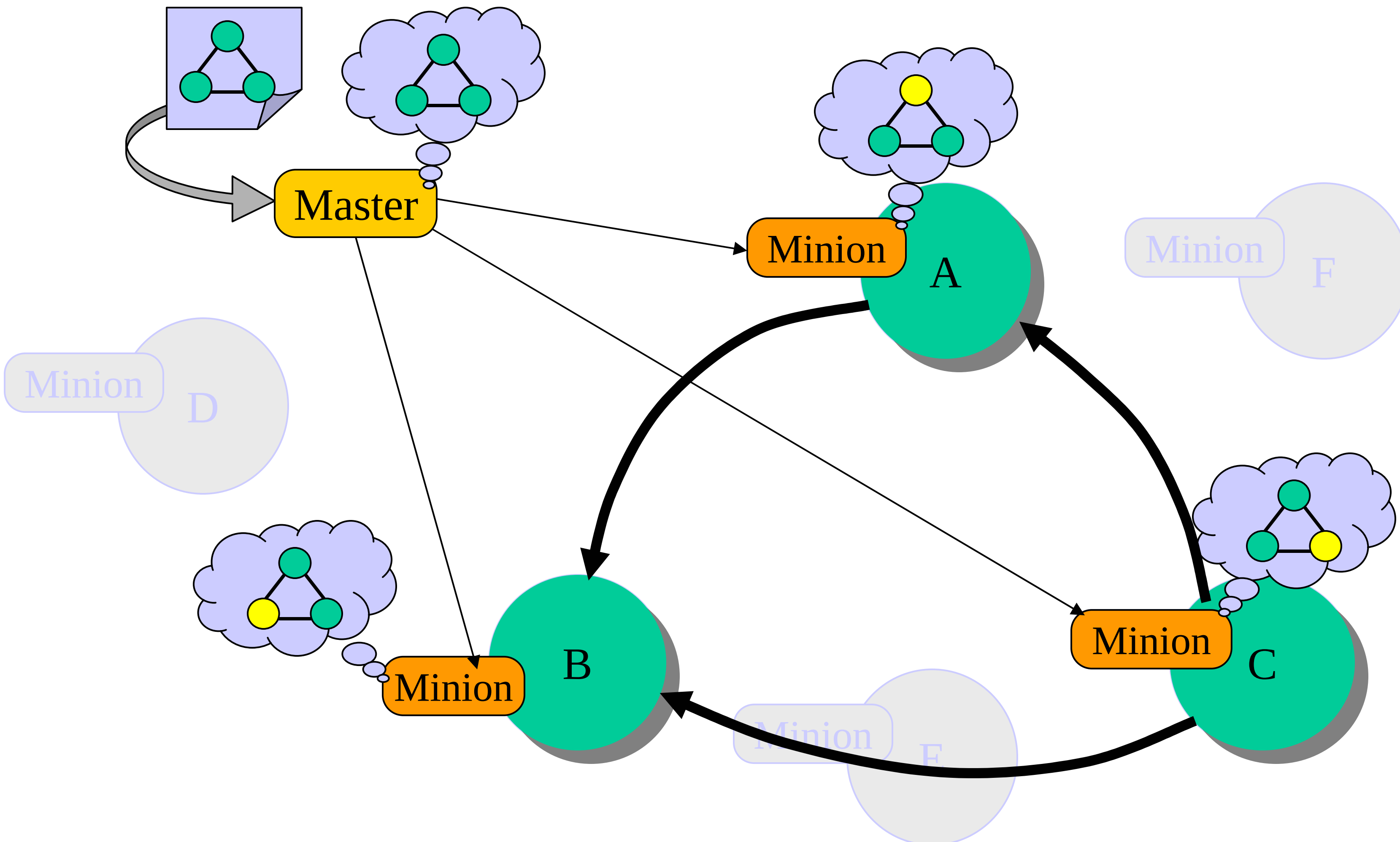



```
Datalink:= { Framing=Ethernet; bandwidth=1g;
                 SourceAddress=%1::vlbid;   DestinationAddress=%2; }
Topo_vlbi_200406 := {
        Correlator:=corr1.usno.navy.mil::vlbid;         // USNO
        DataLink( mkiv.oso.chalmers.se, Correlator );   // OSO Sweden
        DataLink( pollux.haystack.mit.edu, Correlator );// MIT Haystack
        DataLink( ggao1.gsfc.nasa.gov, Correlator );    // NASA Goddard
        }
```

C++ Code invocation example:

```
eVLBI = new ASTDL::Topo( “Topo_vlbi_200406”);  // Get the topology definition
Stat = eVLBI.Create();                          // Make it so!
```

The AST Process



ASTDL Driver Example

```
#include "class_Topo.h++"
#define DRAGON_TCP_PORT 5555;
//
// User prime mover "ast_master" for AST miniond
//
using namespace std;
using namespace ASTDL;

int main(int argc, char *argv[])
{
  int stat;
  Topo *topo;
  topo = new Topo(argv[1]);
  if(topo == NULL) exit(1);
  stat = topo->Resolve();
  if(stat != 0) exit (2);
  stat = topo->Instantiate();
  if(stat != 0)
    { cout << "Error stat=" << stat << endl; exit(3);}
  stat = topo->UserHandoff();
  stat = topo->Release();
  exit(0)
}
```

Read topology definition source and create the Topo object

Resolve hostnames and other service specific data

Establish all the connections

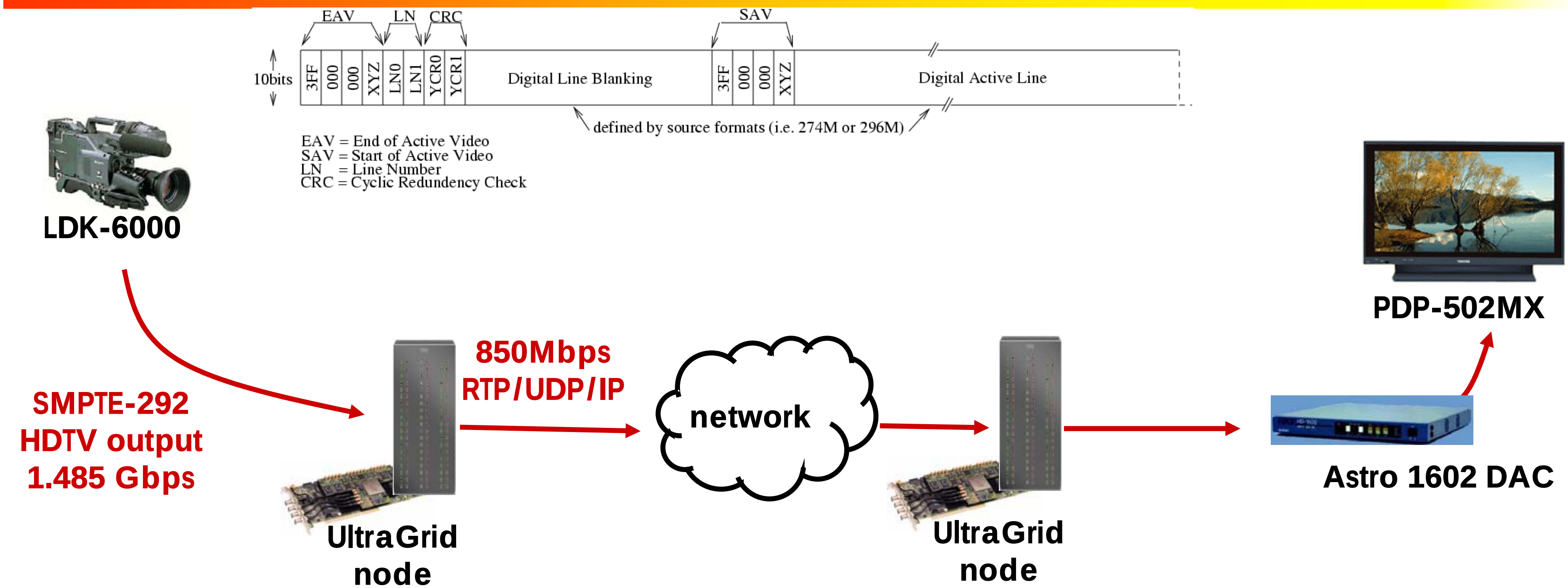
exec() the user and pass off the connections.

All done. Tear it down.

High Definition Collaboration and Visual Area Networking (HD-CVAN)

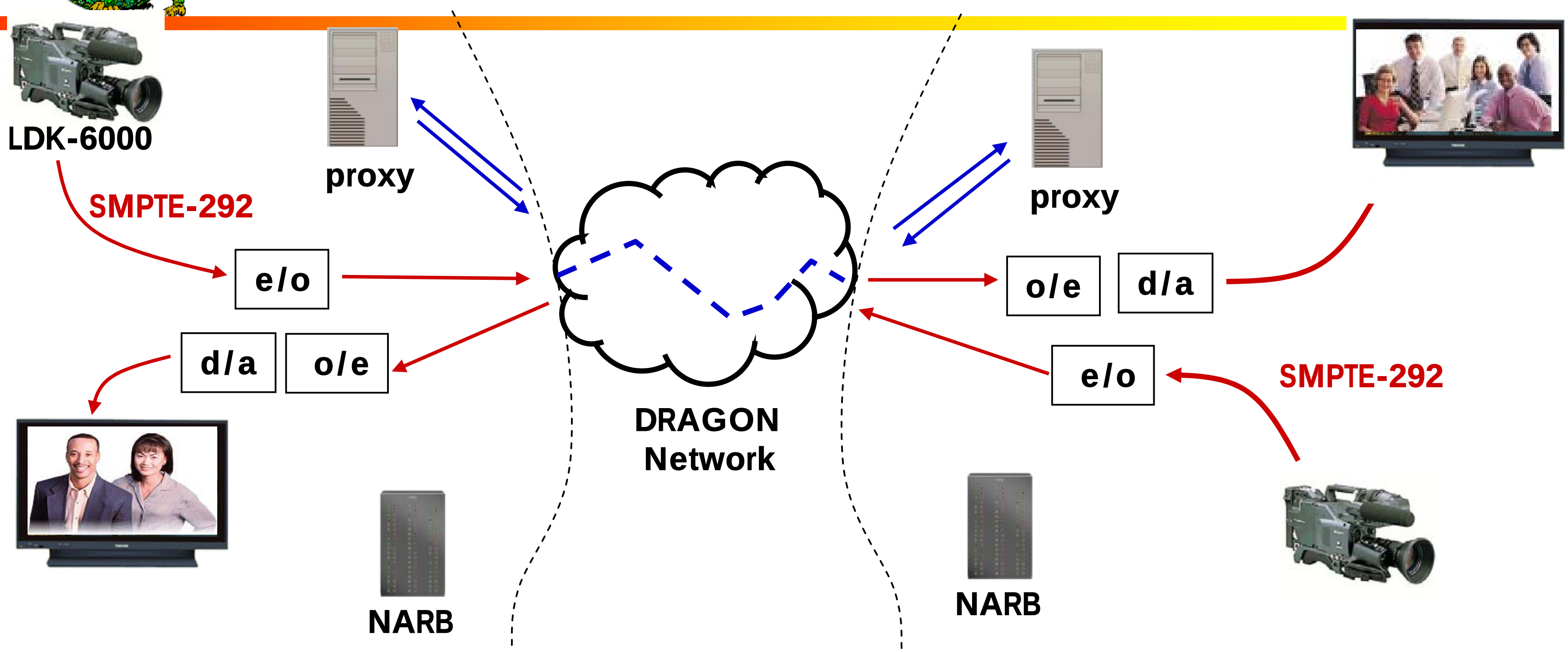
- **Dragon dynamic resource reservation will be used to instantiate an application specific topology**
  - Video directly from HDTV cameras and 3D visualization clusters will be natively distributed across network
- **Integration of 3D visualization remote viewing and steering into HD collaboration environments**
- **HD-CVAN Collaborators**
  - UMD VPL
  - NASA GSFC (VAL and SVS)
  - USC/ISI (UltraGrid Multimedia Laboratory)
  - NCSA ACCESS

Uncompressed HDTV-over-IP Current Method



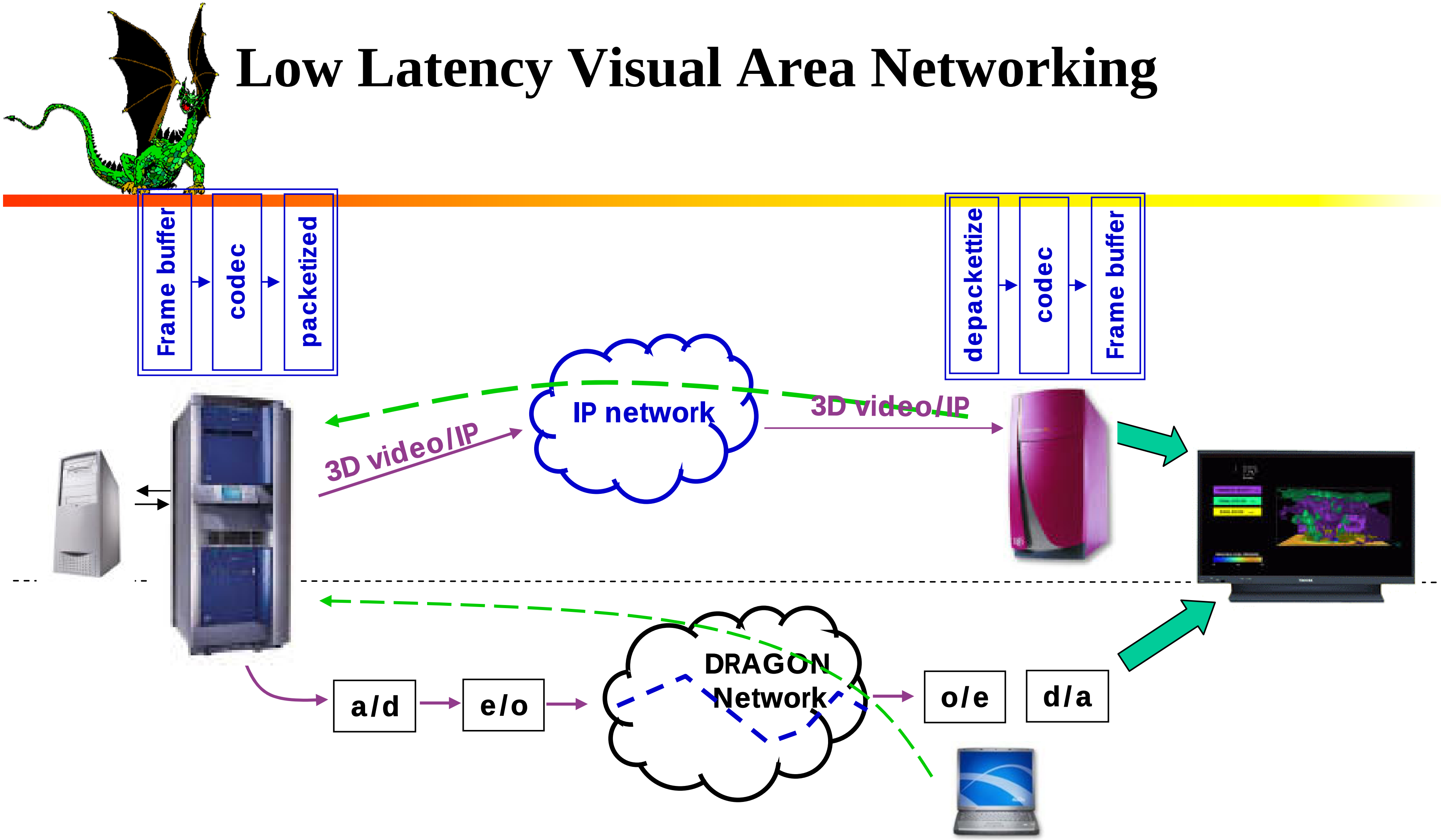
- Not truly HDTV --> color is subsampled to 8bits
- Performance is at the mercy of best-effort IP network
- UltraGrid processing introduces some latency

Low latency High Definition Collaboration DRAGON Enabled



- **End-to-end native SMPTE 292M transport**
- **Media devices are directly integrated into the DRAGON environment via proxy hosts**
  - Register the media device (camera, display, …)
  - Sink and source signaling protocols
  - Provide Authentication, authorization and accounting.

Low Latency Visual Area Networking



- Directly share output of visualization systems across high performance networks.
- DRAGON allows elimination of latencies associated with IP transport.

Status to Date

- **Wavelength Selective Switch operational at CLPK (Univ of Md.)**
  - Installed and operational Spring 2004,
  - Installing Phase 2 ROADMs ->July 2005
- **Initial VLSR functionality demonstrated**
  - Successful interop tests with/across Movaz, Juniper, Sycamore, Ethernet switches. More to come…
  - VLSR being used in other testbeds: CHEETAH, HOPI and USN are interested…
- **Initial NARB demonstrated at SuperComputing 2004 in Pittsburgh**
  - Control Plane group planning to begin inter-domain experiments between Omninet, USN, CHEETAH, and DRAGON.
  - Currently planning optical GMPLS peerings with GIG-EF(wavelength) and HOPI (Ethernet)
- **DRAGON Software Suite Release 1.0Beta scheduled May 1, 2005.**
  - Will include VLSR code (RSVP-TE, OSPF-TE), NARB, integrated PC routines, initial installation and configuration guides, verification/testing tools NUTTCP.
  - FreeBSD and/or Linux
- **HD-CVAN UltraGrid node, using DRAGON technologies, demo'd at SC04**
  - Will be tested in DRAGON network over Spring 05
- **Realtime eVLBI across the DRAGON (without full dynamics) demo'd at SC04**

Future Objectives

- **Complete the DRAGON network deployment**
  - Full wavelength switching capability
  - Including move the Haystack link to 10Gig
- **Re-write the Traffic Engineering DataBase for NARB/OSPF-TE**
  - Include Path Computation attributes for optical OSNR routing
  - Include time domain attributes for scheduling and bookahead
  - More complete AAA capabilities
- **Integration of Common Service Definition model into inter-domain service routing**
  - XML based definition and distribution (experimental – see how it works)
- **XML based AST builder middleware**
  - Generalize the ASTDL to XML format (experimental)
  - Integration with other resource schedulers
  - Integration with other application specifications (e.g. eVLBI EGAE)
- **Experiments and tests**
  - DRAGON GMPLS control plane services over HOPI
  - End-to-end instantiation of eVLBI AST including EU and/or JP links with Haystack and GSFC

For more information…

- **Web:** **dragon.maxgigapop.net**
- **Contact:** Jerry Sobieski
  jerrys@maxgigapop.net
  +1-301-314-6662