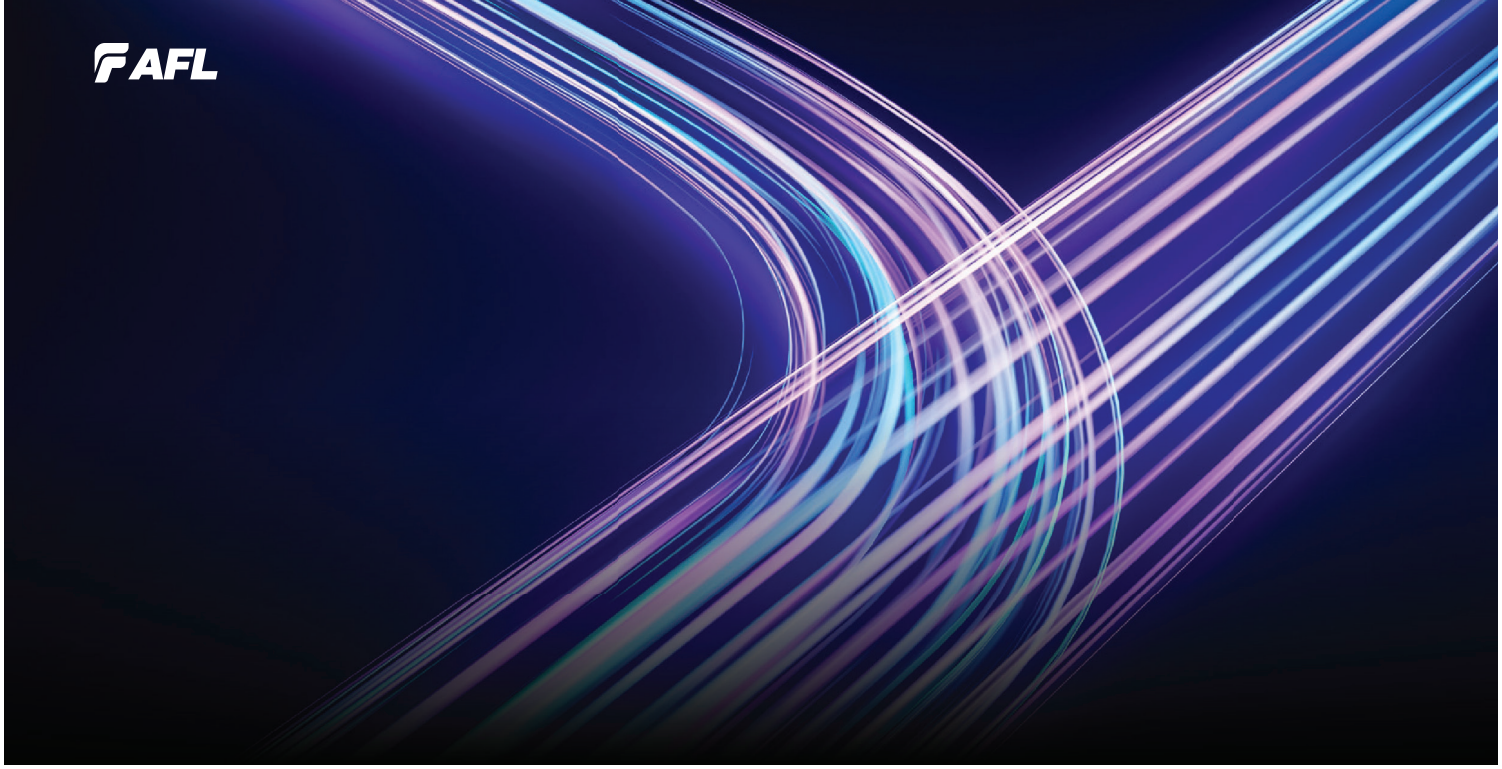
Meshing in AI and Hyperscale Data Centers: Practical Guidance for Evolving Infrastructure Design



Executive Summary

AI and hyperscale data centers require consistent bandwidth and predictable, low latency. A mesh network, defined as a many-to-many interconnect where each endpoint maintains several independent paths to other endpoints, provides these critical attributes.

A mesh operates on two levels: logical and physical. The logical fabric determines how switches direct traffic, while the physical layer, or physical fiber plant, provides the meshed fiber infrastructure.

Mesh deployment requires multiple planes (each of which is an independent copy of the fabric) with distributed links and connections typically established using high-fiber count, pre-terminated multi-fiber assemblies. This approach reduces installation time, optimizes cable weight and volume, improves installation consistency, and supports expansion. The outcome is a network with more available paths, fewer bottlenecks, streamlined maintenance, and simplified, accelerated growth without redesign.

This white paper provides a practical guide to meshed network architectures for technical professionals involved in data center design and expansion, covering meshed network fundamentals, operational impacts, and deployment strategies.

Our objective is to foster better understanding of meshing as a strategy for achieving performance targets and enabling scalable growth. For organizations exploring this approach, direct consultation is available. Our dedicated team can assist with designing and implementing a mesh tailored to specific infrastructure requirements.

Contributors

Dr Alan Keizer
Senior Technology Advisor, AFL

Ben Atherton
Technical Author, AFL

Keith Sullivan
Director of Stategic Innovation, AFL

Paige James
Design Manager, AFL

Jack Kallas
Solution Engineer, AFL

What is Meshing in a Data Center?

Meshing describes an approach to data center network architecture where each node (e.g., server, GPU shelf, storage system) maintains several independent paths to many other nodes. These paths are made possible by the dual layer mesh composition:

Logical Mesh (Fabric)

The fabric defines the switch topology, most often a Clos topology (e.g., a fat tree, which is a type of Clos). This structure creates many equal-cost routes between endpoints – Ethernet networks use equal-cost routing to forward packets to a single destination over multiple paths with the same priority, improving load distribution and redundancy. Traffic is distributed through Equal-Cost Multi-Path (ECMP), which means flows are spread rather than forced into a single path. This approach maximizes bandwidth utilization and prevents any single link from becoming overwhelmed.

Physical Mesh (Fiber)

The physical mesh refers to the fiber plant, which connects nodes to separate planes and switch blocks to ensure diversity in the physical infrastructure. Connections do not converge on a single switch or cross-connect but instead terminate in different locations (often laterally across racks or rows) to reach distinct planes or switch blocks. This multilink cabling resembles a mesh.

"Together, the logical and physical mesh layers eliminate single points of failure while providing the consistent, high-bandwidth connectivity that AI workloads and hyperscale applications demand. When one path fails or becomes congested, traffic automatically flows through alternative routes without service interruption. Additionally, mesh topology enables horizontal scalability, allowing the fabric to grow wider without introducing additional tiers or latency."

Dr Alan Keizer
Senior Technology Advisor, AFL

Why Mesh?
Use Cases and Benefits

Connecting "One to Many"

AI training workloads are collective by design. GPUs must exchange parameters and gradients in synchronized steps, and a mesh provides each node with multiple usable paths. This design avoids hot spots and reduces tail latency (i.e., the delay experienced by the slowest packets). Because these workloads are tightly synchronized, even minor network delays can stall the training process.

Consistency is often more valuable than peak performance, since a set of parallel links delivers steadier throughput than a single, large link. Correctly deployed, meshing resilience becomes inherent, as the network can continue to function even if a link, a switch, or an entire plane becomes unavailable.

Front-End and Backend Networks

Meshing supports both the backend network (BENW) and front-end network (FENW). On the back end, the high-bandwidth, low-latency fabric connecting GPUs or accelerators depends on meshing to execute collectives such as all-reduce and all-to-all:

- **All-reduce –** aggregates values across devices. Used when every GPU needs the same combined result, such as summing gradients during training.
- **All-to-all –** redistributes data so that each device sends and receives unique portions to and from every other device. Used when GPUs need to exchange different pieces of data, not just share the same result.

For more information, see our white paper:
AI Data Centers: Scaling Up and Scaling Out White Paper.

On the FENW, where clients access services and east–west flows are less tightly synchronized, meshing provides advantages as well, although the BENW remains the most critical consideration.

Tier-to-Tier Flow

A typical backend network (BENW) fabric is built in layers. The first layer comprises leaf switches (sometimes called edge or T0 switches), which connect directly to endpoints such as GPUs or accelerators. Each endpoint usually connects through two or more uplinks, with each uplink pinned to a different network plane. Above the leaf layer are spine switches (also called aggregation or T1 switches). Each leaf connects to multiple spines, and in very large fabrics, a third layer of core switches (T2) may be added, which interconnect the spines. With this hierarchy in place, any GPU or accelerator can typically reach another in just two or three hops.

Planes: The Multiplier

A plane represents a complete instance of the fabric. Each plane comprises multiple self-contained elements (e.g., T0 switches, T1 switches, routing state, failure domain). Endpoints connect to two or more planes using separate ports or Network Interface Cards (NICs), or by allocating separate lanes within a single port or NIC.

If one plane provides k equal-cost paths, P planes provide approximately P × k distinct paths without shared links. This design allows operators to maintain or drain one plane while traffic continues across others. Job placement strategies can assign workloads to a subset of planes or stripe them across multiple planes to maximize bisection bandwidth, defined as the total capacity available between two halves of the network. Higher bisection bandwidth directly translates to lower latency and more efficient AI training.

Techsplainer: Planes and Paths

P = The number of planes (independent highway systems / independent network fabrics).

K = the number of paths (routes) within a plane.

One Plane = One Highway System
A single plane can be thought of as one highway system with multiple routes.

Multiple Planes = Multiple Independent Highway Systems
For example, three planes comprise three separate highway systems:
- One plane = 4 routes (or one plane = 4 paths between two computers).
- Three planes = 12 total routes (or three planes = 12 distinct paths between those computers).

Why "Without Shared Links" Matters
In networking, if one plane fails, traffic can be rerouted seamlessly through the remaining planes.

While meshing provides clear benefits in Ethernet-based fabrics, InfiniBand follows a different design model. The switched fabric topology with point-to-point bidirectional links delivers scalable, high-bandwidth connections without relying on meshing. In practice, meshing is both less necessary and more difficult to implement in InfiniBand environments. A full comparison of InfiniBand and Ethernet lies beyond the scope of this paper, but the trend in large-scale AI deployments shows Ethernet emerging as the preferred network protocol.

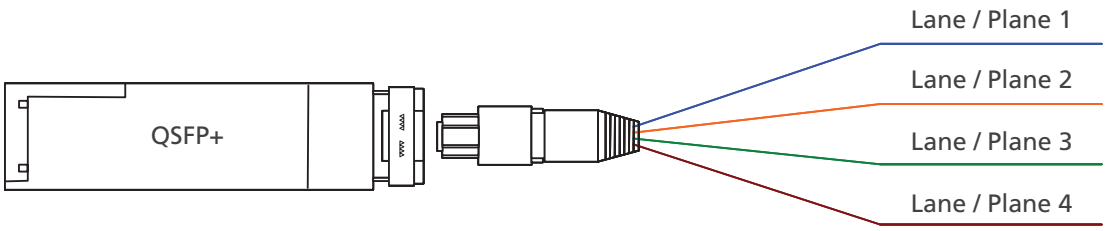
How Meshes Are Built

The logical concept of a mesh is straightforward. The physical implementation determines whether the design works reliably on day one and remains manageable (or 'debuggable') on day one thousand. This section explores several common build patterns that shape how meshes take form in real environments.

The Physical Concept of Meshing

Today's larger AI clusters operate at port bandwidths of 400 Gb/s or 800 Gb/s, with leading-edge deployments now being built around 1.6 Tb/s network architectures. These bandwidth requirements drive specific connector and cabling choices that determine how mesh networks can be physically implemented.

- **400 Gb/s networks** typically use 400GBASE-DR4 transceivers with MPO-8 connectors, transmitting data as four lanes of 100 Gb/s. Each lane uses a dedicated pair of fibers, and the eight-fiber MPO-8 connector carries all four pairs, each delivering a 100 Gb/s optical signal.
- **800 Gb/s networks** use 800GBASE-2xDR4 transceivers, which split eight lanes of 100 Gb/s across two MPO-8 connectors.
- **1.6 Tb/s networks** follow the same architecture as 800 Gb/s, but increase the lane speed to 200 Gb/s.



In a mesh network, the lanes from an individual transceiver on one end of the link are meshed to individual planes on the other end. In a four-plane mesh network, the four lanes from each of four transceivers on end A mesh so that all four Lane/Plane 1s go to a "Plane 1" transceiver on End B. Likewise, the four Lane/Plane 2s go to the Plane 2 Transceiver on End B, following the same pattern repeating across all planes.

In practice, the ports on one end of the link (on the left in the example above and below) are generally located on the same device while the ports on the other end will be physically separated on different devices, which could be in the same rack, in the same row of racks or in completely different rows.

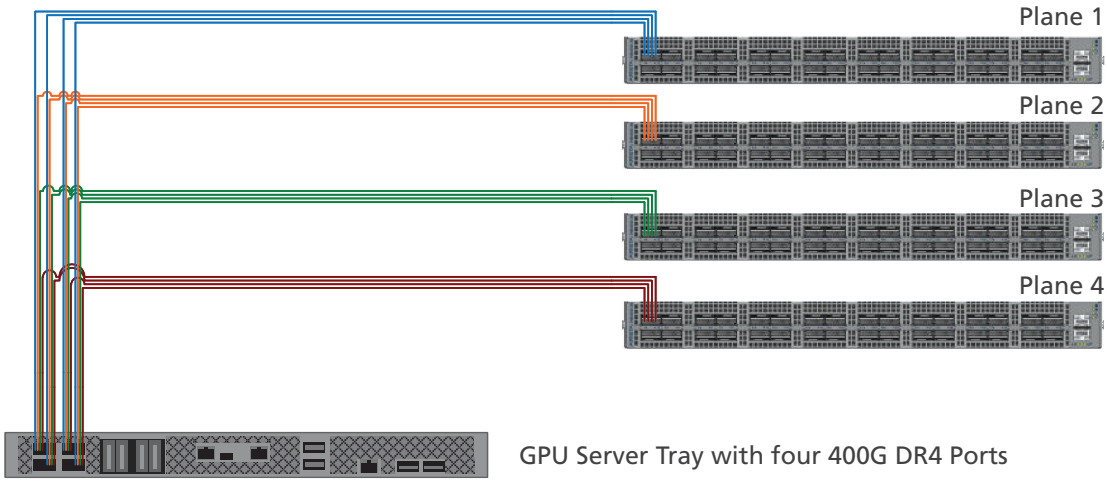
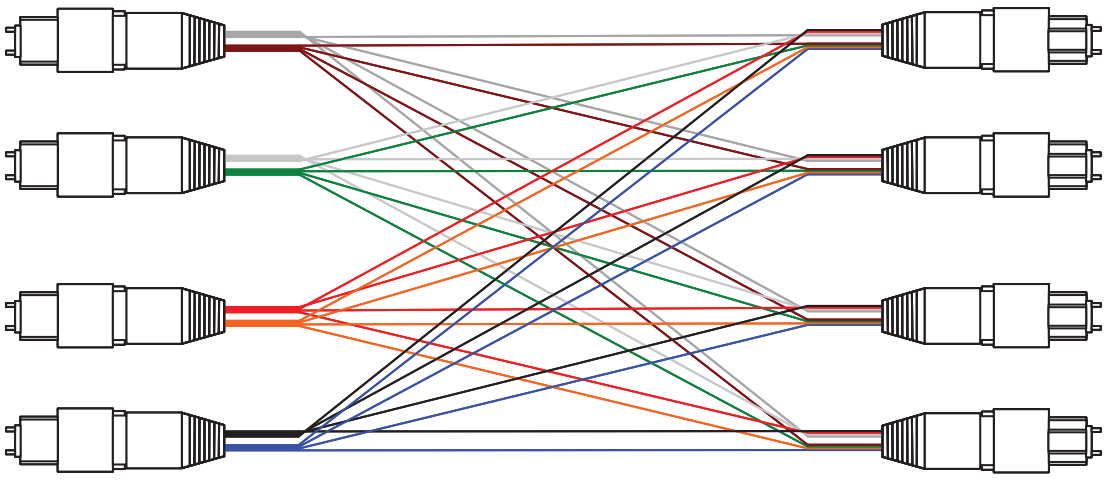


Image: GPU Server Tray with 4X 400GBASE-DR4 ports, each with four optical lanes of 100G, connecting to 4 physically separated T0 switches. Switches could be in the same rack, same row or different rows.

Meshing is conducted on all links between the GPU servers and the T0 switches, and between the T0 and T1 switches. There are three fundamental methods of constructing a mesh and a variety of methods to deploy these meshes in the data center.

Constructing the Mesh



A Simple 32f Shuffle Patchcord

32 fibers are commonly terminated into four MPO connectors on End A. On End B, the fibers are shuffled as per the above diagram and terminated into the four MPO connectors on End B. This is the simplest form of a mesh and can be deployed between a patch panel and a switch in the same rack (akin to a standard top of rack patch panel). The patchcords can be manufactured at standard lengths and held in inventory for immediate deployments.

A Mesh Cassette / Patch Panel

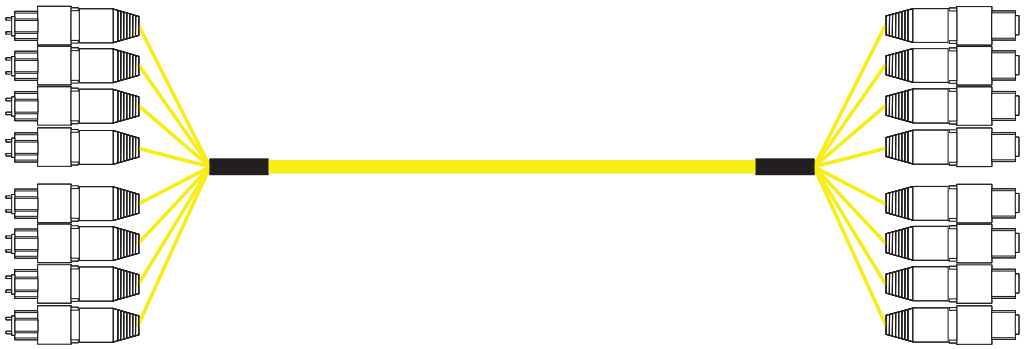
The 32f shuffle assembly is constructed as above and plugged into a shuffle cassette or a patch panel. The standard trunk MPOs are plugged in on the back side, shuffled, and patched from the output MPOs on the front directly to the switch. This is a simple method, which allows the use of standard trunks and patchcords, with the mesh confined to a cassette, panel, or box. However, this method consumes more space in the data center and adds an extra connection (and resulting loss) into every link.

A Simple 32f Shuffle Patchcord

Similar to a shuffle patchcord, groups of 32 fibers can be meshed between sets of four MPO connectors on either end of the trunk. Fiber counts of the trunks can therefore only be in multiples of the 32f mesh, giving common trunk fiber counts of 32f, 64f, 96f, 128f, 192f & 256f.

While a shuffle trunk can be deployed to provide a direct connection between a GPU rack and a T0 switch, or between T0 and T1 switches, each trunk must be custom-made, and the complex design limits flexibility for future moves, adds, and changes. In contrast, shuffle patchcords deliver the same connectivity benefits with a simpler, more adaptable deployment model.

High Count Patchcord



A cross between a shuffle patchcord and a shuffle trunk, a high-count patchcord combines two or more 32f mesh units into a single patchcord type assembly. A trunk patchcord could be deployed in the same rack or in the same row of a data center but should not be used as main infrastructure running down the length of the data center.

Mesh Implementation Scenarios

The way meshing can be implemented into a large-scale data center can be viewed in (but not limited to) various common architectures.

Striping: GPU to T0 and T0 to T1

Large-scale AI network architectures commonly deploy the NVL72, NVIDIA's latest GPU rack. Each NVL72 contains 72 NVIDIA Blackwell Ultra GPUs, distributed across 18 servers, with 4 GPUs per server. These GPUs handle core AI computations in the data center. Within a single NVL72 rack, GPUs are meshed using NVIDIA's NVLINK, forming a scale-up network. Multiple NVL72 racks installed in a row must be meshed at two levels:

- **T0:** interconnecting racks within the same row
- **T1:** interconnecting racks across the entire data hall

A common method for achieving this multi-level interconnectivity is striping.

Each server includes four OSPF ports, resulting in 144 MPO connections per rack. Meshing GPUs at the T0 level involves aligning ports into a lane of T0 switches. For organizational clarity, ports can be assigned color identifiers (e.g., port 1 = blue, port 2 = orange, port 3 = green, port 4 = brown). Before connecting to the T0 switches, MPOs are grouped within a patch panel housed in the rack:

- Tray 1 contains all 18 port 1 (blue) MPOs
- Tray 2 contains all port 2 (orange) MPOs – pattern continues for ports 3 and 4

This striping configuration is replicated across all NVL72 racks within the row, feeding into the corresponding T0 switch infrastructure. Once all NVL72 racks are patched into the T0 switch infrastructure, GPU meshing can begin. Using shuffle patch cables (also referred to as shuffle cables) each port 1 (blue) MPO from every NVL72 is routed into a dedicated channel, enabling direct communication between GPUs across the row. A shuffle patch cable is a type of shuffle cable used in the first equipment connection, typically linking a node (server) or switch to a panel. This helps establish the logical mesh within the channel, which is the full equipment-to-equipment connection. A link is a segment of that channel, and the mesh can be logically created in any or several of the links. Horizontal meshing is typically implemented across the T0 switches, with each switch assigned to a specific channel.

With all GPU rows striped and meshed at the T0, the next step is to extend this striping and meshing at the T1. Each T0 rack is patched into the T1 infrastructure using a similar method as before. Specifically, each T0 is connected horizontally to a T1 patch panel, with each T0 rack assigned a dedicated tray. This provides clear separation between T0 rack 1, T0 rack 2, T0 rack 3, and so forth.

Using shuffle patch cables to interconnect to the T1 switches enables full GPU meshing across rows. Unlike T0 horizontal meshing, T1 meshing is typically implemented vertically. Starting from the first port of each tray, each lane is shuffled into dedicated channels. Each row of T1 switches corresponds to a specific port in the patch panel, maintaining alignment with T0 racks.

GPU to Patch -> Shuffle Patch to T0 -> Leaf to Patch -> Shuffle Patch to T1

This method is typically preferred for network meshing because none of the connections rely on permanent links. Shuffling is achieved using high-count shuffle patchcords that connect multiple GPUs within a row, with those GPUs subsequently meshing with the remaining GPUS at T1.

GPU to T0 -> Shuffle Trunk to Patch -> Patch to T1

In this scenario, a patchcord connects the GPU to a T0 switch, while a shuffle trunk* bridges T0 to T1 patch panel. A patch cable connects the patch panel to the T2 switch. While this highly customizable approach reduces cables counts, the shuffle trunk would function as a permanent link, meaning future adjustments would require the costly and disruptive replacement of the full link (i.e., the gain in flexibility could also potentially increase expense).

* ‘Mesh’ and ‘shuffle’ may be considered interchangeable when applied to a cable assembly, panel, or cassette.

Direct Patching (Rack-to-Rack or Rack-to-Row)

In this approach, pre-terminated multi-fiber trunks (e.g., MTP/MPO) or newer compact connectors (e.g., MMC) are run laterally between racks and landed in plane-specific switches. Direct patching shortens paths, reduces congestion in main pathways, and allows fast installation using factory-tested components with verified losses and polarity. Without strict labeling and plane separation, however, cabling can become difficult to manage, and very large sites usually adopt a cross-connect layer to preserve flexibility.

This type of meshing is more common in a smaller network where not as much customization is required. This is because direct patching requires many more cables since the meshing is being done going from higher count MPO cables to lower count LC cables (as opposed to meshing inside of a trunk or shuffle box). An example of this would be having an MPO-8 fiber cable connecting from a GPU to a MPO to LC fanout cassette. From here, you can align the mesh using LC connectors to another MPO to LC fanout cassette.

Intermediate Cross-Connect (Row or Room ODF)

Here, trunks from racks terminate on an Optical Distribution Frame (ODF) for each plane, with subsequent patching to spines or other rows. This design provides clear change control and well-defined 'jump-off' points for rerouting or expansion, and also simplifies fault isolation, since an operator can view an entire plane's layout at once. The trade-off is a small increase in optical loss and cost from the additional patch point.

Equipment Panels with On-Panel Patching

Leaf and spine cabinets often include front-accessible panels or cassettes that normalize multi-fiber ports, enabling plane-by-plane patching without opening switch enclosures. This design improves serviceability and ensures plane identification is visible at the working face. Success depends on careful documentation of port maps and polarity throughout the system.

Shuffle Modules or Shuffle Boxes (Multi-Fiber Re-Map)

Passive cassettes that redistribute fibers from one multi-fiber connector to several others. These modules are commonly used to implement plane striping, such as dividing a 16-fiber link into four groups that connect to four separate plane trunks. Shuffle modules also align transceiver pinouts with structured cabling (e.g., when supporting 2×400G DR4 breakouts), and provide a clean, repeatable method for link distribution (but require precise documentation to prevent troubleshooting complexity).

Shuffle Cables (Pre-Mapped Multi-Fiber)

A shuffle cable follows the same principle as a shuffle module but implements the mapping inside a fixed cable with multi-fiber connectors on each end. Factory mapping ensures fast deployment and consistent polarity, making shuffle cables well-suited for fanouts or plane distribution in confined spaces. If the mapping needs to change later, these cables offer less flexibility than panel-based shuffles.

Design Considerations for Reliable Meshes: What Makes Meshes Viable in Real-World Applications?

Design Clarity: Stripes and Striping

Mesh connections can be dense and cable rich. Best practice mandates clear and consistent layout and labeling. Mesh stripes should be logical and visual. Uplinks should be distributed across planes to avoid concentrating critical paths on a single physical pathway or power domain. Technicians should be able to readily identify a port's plane assignment; while labeling is a common best practice, additional identification methods such as RF ID and color coding may also be used. Any approach that enhances clarity and supports consistent layout is considered good practice.

Easy, Fast Installs: Pre-Terminated Assemblies

Ease and speed of installation is essential for large fabrics. Pre-terminated assemblies such as trunks, breakout cables, and shuffles reduce on-site. Cable assemblies should be designed with tail lengths to match port positions and may span multiple racks. Pre-terminated assemblies deliver predictable optical loss budgets and managed polarity, shortening installation windows, and simplifying change control.

Techsplainer:

Trunk Cables
Pre-terminated assemblies designed to connect data center zones or floors. Trunk cables streamline installation and support scalable infrastructure.

Breakout Cables
Fan-out cables with multiple legs tailored to port layouts. Used to break out trunk cables, connecting high-density patch panels to active equipment.

Shuffles
Specialized assemblies that re-route or "shuffle" port connections to meet specific polarity or port assignment requirements. Ideal for managing custom routing and port mapping.

Maintainability

Maintainability follows from structure:

- Plane structured fabric provides redundant paths and limits the blast radius of any failures.
- Consistent hop counts and symmetric wiring reduce hidden micro-hotspots and unexpected latency, making network tuning much easier.
- Well configured mesh design clearly translates the logical network to the physical network, improving speed and accuracy of buildout and maintenance work.

Multi-Fiber Connectivity: Count and Density

Multi-fiber connectivity should be standardized within a site. Most AI deployments rely on 8, or 16-fiber links. Multiple links may be aggregated into high count cables. Standardizing on counts and mesh configuration per network layer at a site simplifies operations. Compact multi-fiber connectors (such as MMC) enable far higher port density per rack unit than traditional MPO designs making fiber management a critical design consideration. For large rooms, long single-mode fiber runs are typical, while short copper or AOC alternatives may prove impractical. Planning should account for transceiver evolution, ensuring clean mapping from 400G to 800G to 1.6T optics.

Polarity, Polarity, Polarity

Polarity control is critical. The send/receive orientation of fibers must be managed end-to-end, across every panel, trunk, shuffle, and patchcord. A single polarity standard should be chosen and enforced. 'Flips' in the field should not be standard practice; the design or manufacturing should be corrected to assure correct polarity as installed. Mismatched polarity can result in signal loss or complete communication failure between devices. Poor polarity control can also complicate future moves, adds, and changes, risking downtime.

Standards vs Custom

Standard components such as trunks, cassettes, and shuffles shorten lead times, simplify spares, and support multi-vendor sourcing but may not provide optimal tail lengths for high density management and may need additional connector interface to realize the mesh fabric design. Custom cable assemblies, particularly shuffle cables can provide then needed fiber mapping with optimized tail length, fewer components per channel and fewer connector interfaces for better optical performance. In practice, large AI data centers provide scale to support custom design and often benefit from optimal design and optical performance.

Integrating Logical and Physical Design

Successful meshes require logical and physical alignment. Start by defining planes (initially as few as two, expanding to four or more as clusters grow) while keeping each plane consistent in hop count and switch type. Each node has one uplink per plane to enable traffic striping and fault tolerance. Connector families should be fixed for at least two future speed transitions. Fiber plants are designed per plane, with dedicated patchcords, trunks, and panels. Shuffle modules or shuffle cables can be used wherever lane redistribution is necessary (e.g., DR4/DR8 breakouts or plane striping), avoiding ad hoc field work. Finally, each plane should be monitored and tested as a separate network.

The result is a mesh that remains both scalable and operable because the structure is easy to see and easy to maintain.

Where Meshes Are Going (Next 3–5 Years)

AI clusters continue to grow in both size and synchronization, with network demands rising accordingly. Several trends define the next phase of meshing in data centers. These will impact the use and design of meshes in the network

Connection Counts and Density

Nodes will require more uplinks or higher lane rates to maintain bisection bandwidth in step with compute growth. Multi-fiber technology will advance toward denser formats such as MMC, reducing panel footprints while increasing port density. This will require mesh components – modules and cables – fitted with the new connector types and will put more emphasis on layout and good practices. Higher fiber counts will be needed throughout the fabric at all levels.

Higher-Count Multi-Fiber

As optics transition from 400G to 800G and 1.6T, fiber lane counts and mappings will evolve. DR4 and DR8 families are expected to remain widely used for short-reach single-mode applications inside data halls. DR8 and Base16 links will need 8x8 mesh solutions in lieu of the 4x4 meshes used with DR4 / Base8 connectivity. It will also be important to plan migration to 16 fiber connectivity in cables, modules and panels. to ease migration. Higher bandwidth will put more stress on the optical power budget, favoring simpler channels with fewer intermediate connectors. Shuffle cables provide lowest channel optical loss.

Co-Packaged Optics (CPO)

Co-Packaged Optics (CPO) place optical engines adjacent to or directly on the switch ASIC, reducing electrical reach and potentially lowering power consumption. If CPO adoption becomes mainstream, operators should expect shorter patching distances from switch face to panel (or even direct fiber egress from the chassis). Service models may shift, with more replacements occurring at the panel or cassette level rather than through individual transceivers. Also, as optical density increases, rack-level thermal and fiber management will grow in importance. In some applications, the mesh may move inside the chassis and generally fiber counts and density will continue to grow.

Optical Circuit Switching (OCS)

Some organizations are exploring Optical Circuit Switching (OCS) to create temporary, optical circuits between endpoints with heavy traffic. OCS does not replace the packet fabric but rather augments the fabric for specialized use cases such as skewed but predictable training flows or rapid reconfiguration between job phases. OCS is now used as a T3 switch (spine) in some large AI clusters and is expected to find application in DCI and transport layers as well. In some applications, the OCS can be used to form the meshed interconnect needed in a fabric.

Closing Takeaways

Meshing is often viewed as a complex concept, but a successful mesh design depends on a handful of disciplined practices that make networks predictable and resilient. The lessons are practical, not theoretical.

Key takeaways:

- Mesh logical and connectivity fabrics are used extensively in large, Ethernet based AI training and inference clusters.
- A mesh is not a single feature but the combination of multiple planes and carefully managed physical wiring.
- The physical mesh can be realised multiple ways with the best choice dependent on application. For large AI clusters, meshes built using mesh modules in panels and shuffle cables are recommended.
- Shuffle cables can provide the most compact solution with the lowest optical loss however these may require custom design.
- Pre-terminated multi-fiber assemblies accelerate installation and simplify maintenance.
- Polarity should be standardized early and documented rigorously; inconsistency here remains one of the most common and disruptive sources of installation and maintenance delays.
- Future planning is essential. Every design should anticipate the next plane and the next speed tier. Decisions made today will determine whether scaling is smooth or disruptive tomorrow.

A well-constructed mesh remains easy to operate because the overall design is easy to understand, even as density, optics, and architectures evolve.

Founded in 1984, AFL is an international manufacturer providing end-to-end network solutions to the energy, service provider, enterprise, hyperscale and industrial markets. The company's products are in use in over 130 countries and include fiber optic cable, assemblies, and hardware, transmission and substation accessories, outside plant equipment, connectivity, test and inspection equipment, fusion splicers, and training. AFL also offers a wide variety of services supporting data center, enterprise, wireless and outside plant applications.

Headquartered in Spartanburg, SC, AFL has operations in the U.S., Mexico, Canada, Europe, Asia and Australia, and is a wholly owned subsidiary of Fujikura Ltd. of Japan.