PeSIT Protocol: Concepts, Features, and Implementation

Introduction

PeSIT is a file transfer protocol that stands apart from popular alternatives like **FTP, SFTP,** and **HTTP**. Its design is rooted in interoperability across heterogeneous environments, which makes it conceptually different and more complex if approached through the lens of typical client/server protocols.

PeSIT is best understood not through analogy with existing protocols but by appreciating its foundational design principles.

Peer-to-Peer Partner Relationship

Client/Server vs. Peer-to-Peer

- In **client/server architecture**, services are predefined and fixed on the server side.
- PeSIT operates in a **peer-to-peer** mode, meaning either party can initiate the transfer, and the protocol session is negotiated dynamically.

Analogy

Imagine receiving a phone call: You do not know who is calling, in what language the conversation will happen, or the subject of the call. You must first identify each other and then define the context.

This is the **PeSIT method**: partners agree on their identities, transfer direction (send/receive), and the session purpose before the actual file transfer starts.

Key Implications

- Relationships between each pair of partners must be defined explicitly.
- The same relationship cannot be reused across different machines or transfer scenarios.
- This is more flexible than client/server, but also more configuration heavy.

Protocol Architecture and Features

PeSIT was designed with the **OSI 7-layer model** in mind and primarily operates at **layers 5 through 7** (Session, Presentation, Application).

Error Control

- Uses a **CRC (Cyclic Redundancy Check)** mechanism for each Protocol Data Unit (PDU).

- Now largely obsolete due to modern TCP/IP reliability.

Sliding Window Mechanism

- Manages **flow control** by allowing the sender to transmit multiple PDUs before requiring acknowledgment.
- Also defines **restart points**, ensuring transfers can resume after interruptions.

Bridge/Hub (Store-and-Forward)

- Enables **indirect delivery** through an intermediate node.
- Useful when a direct connection is not possible between sender and recipient.

Note: PeSIT does **not** manage or configure lower network layers (e.g., TCP/IP). All such configurations must be managed externally.

Logical File System

To support cross-platform interoperability, PeSIT introduces logical **file abstraction**.

How It Works

- Files are **not transferred directly** by path or location.
- Instead, logical names are used (e.g., TXT) and mapped to physical paths on each system:
    - On Sender: TXT → C:\test\toto.txt
    - On Receiver: TXT → /tmp/ascii.o

Structured File Format

PeSIT files are **record-based**, not stream-based, and have structured attributes:

Attribute	Description
Name	Logical identity of the file
Format	Fixed or variable-length records
Max Record Length	Maximum bytes per record
Data Encoding	ASCII, EBCDIC, or Binary
Organization	Sequential, Relative, or Indexed

Each partner must map logical PeSIT files to their respective physical file system.

5. Built-In PeSIT Features

Online Data Compression

- Designed for text files with repetitive structure (e.g., banking files).
- Reduces I/O operations and can improve performance in some environments.

Transfer Restart

- Seamless recovery using synchronized restart points if interrupted.

Data Integrity

- Partners verify the number of bytes and records post-transfer.
- Enhanced by (now optional) CRC checks.

Non-Repudiation

- Once a transfer is complete, **neither party can deny it**.
- This mutual guarantee is built into the protocol.

Advanced Capabilities via Transfer CFT

These are **not part of the core PeSIT protocol**, but are available through **Transfer CFT**, Axway's PeSIT implementation.

User-Level Transfer Acknowledgement

- Send a custom confirmation message after post-transfer processing.
- Act as a new message-type PeSIT transfer.

Remote File Naming

- The initiator can request a specific file name on the remote system.
- The recipient must be configured to honor this request.

Transcoding

- Converts encoding formats (e.g., ASCII ↔ EBCDIC) during transfer.
- Some implementations support iconv-based transformations.

File Group Transfer

- Although PeSIT does not support “file groups” natively, Transfer CFT allows grouping multiple files into logical units for transfer.

Summary

Feature	PeSIT Protocol	Transfer CFT
Peer-to-peer session model	✅	✅
Sliding window flow control	✅	✅
File transfer resume (restart)	✅	✅
Compression for structured text	✅	✅
Logical file system abstraction	✅	✅
Custom transfer acknowledgements	❌	✅
Remote file name control	❌	✅
File group handling	❌	✅
Data transcoding	❌	✅

Final Thoughts

PeSIT’s architecture prioritizes **reliability**, **negotiation-based transfers**, and **platform independence**. While it introduces complexity compared to simpler client/server protocols, its robustness and flexibility make it ideal for enterprise-grade, secure file transfers—especially in **regulated industries like banking and finance**.