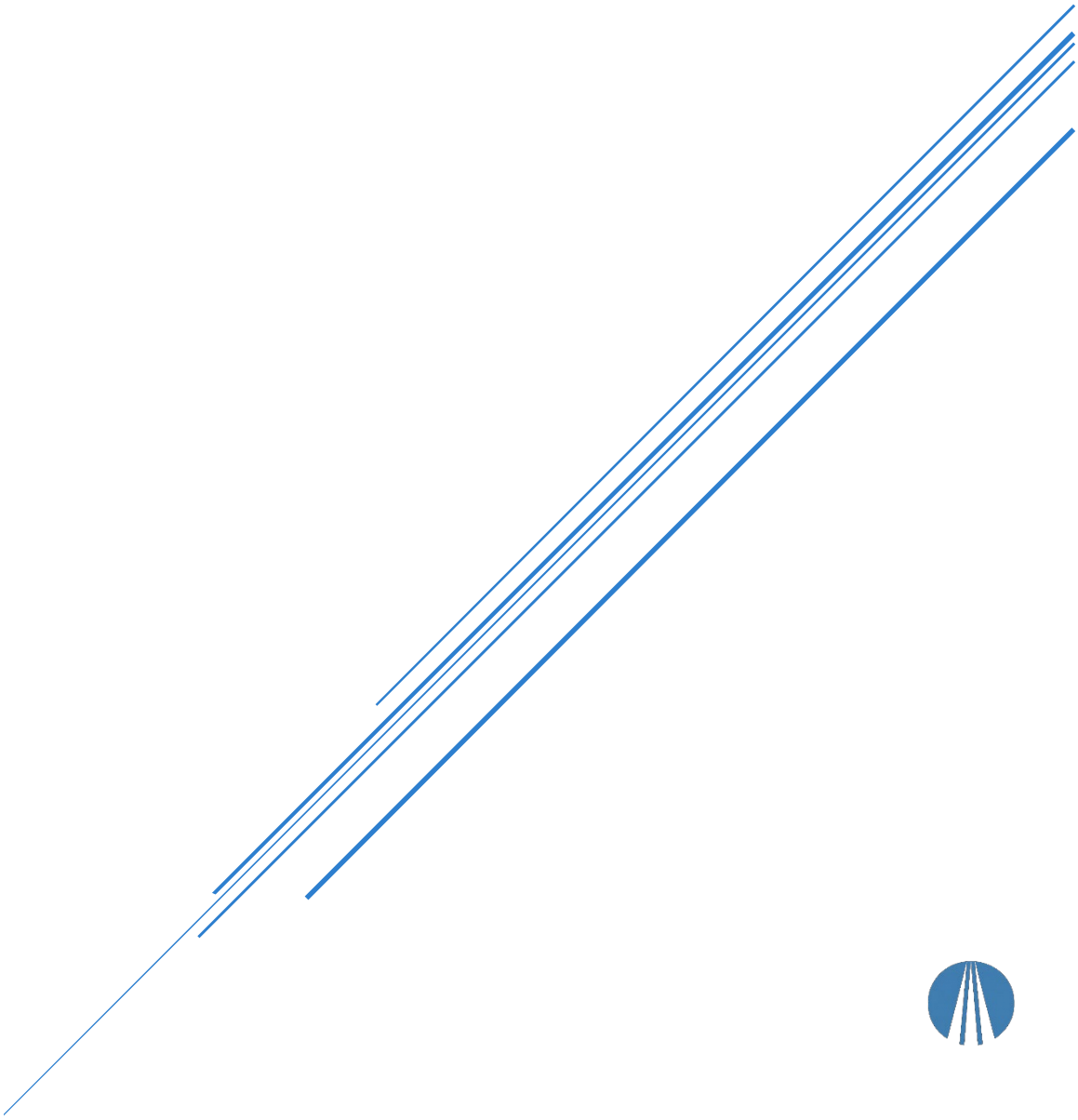
LTI INTEGRATION INCIDENT ANALYSIS

Centralized vs Decentralized Access



CLARITY SYSTEMS CONSULTING

Case Study | Systems & Data Clarity

The Situation

An external learning tool was reported as unavailable within the LMS.
From a user perspective, the issue appeared straightforward:

- The tool could no longer be accessed
- Existing workflows were disrupted
- It appeared as though the integration had failed

At first glance, this suggested a system-level issue.

Understanding the Problem

Initial checks showed that the integration itself was still configured.
There was:

- No indication of platform failure
- No removal of the tool at the system level

This raised a different question:
If the system is intact, why does the tool appear unavailable?

Investigating the Behavior

The investigation focused on how the tool was actually being accessed.
It became clear that:

- Access depended on links created within individual courses
- These links were not centrally managed
- Usage patterns varied depending on instructor setup

A key event was identified:

- A course-level access point had been removed

This removal eliminated the path users relied on to launch the tool.

What Became Clear

The issue was not caused by a system failure.
It was the result of **decentralized configuration**.

- The integration existed at the system level
- Access relied on individually created course links
- Removing a single link removed access within that context

In practical terms:
The system was available — but the access path was not.

Where the Real Risk Was

This revealed a broader issue:

- Inconsistent configuration across courses
- Lack of centralized access control
- Dependence on manual setup

This created:

- fragile access patterns
- inconsistent user experience
- increased support dependency

The problem was not technical failure — it was structural.

Decision Direction

The focus shifted from restoring access to stabilizing the system.
The recommended approach was:

- Establish a centralized access method
- Remove reliance on manual link creation
- Standardize how the tool is accessed across courses

This ensured consistent availability regardless of individual configuration.

Outcome

After implementing a centralized deployment model:

- Access became consistent across courses
- Dependency on individual configuration was removed
- User experience stabilized

The issue was resolved without changes to the underlying integration.

What This Work Reveals

This case highlights a key principle in system design:
Availability depends not only on integration — but on how access is structured.
A system can be technically functional and still appear broken if access is not consistently managed.
Understanding that distinction is critical when diagnosing system issues.