Industrial Data Xchange
The Industrial Data Communications Experts

CASE STUDY PART 1: Integrating a real-time data storage solution with an existing plant infrastructure.

Solution: IDX Nexus Historian used in storage of plant control system data.

Industry: Mining

Benefits:

- ✓ Simple and economical
- ✓ Suitable for all industries
- ✓ Open and extendable
- ✓ Standards-based

Technology used:

- ✓ IDX Nexus Historian
- ✓ Microsoft SQL Server

The IDX Nexus Historian is a powerful data logging and visualisation framework for plant data integration and performance monitoring.

It supports several data compression algorithms that handle offline data buffering and can scale from a few to many thousand tags per second.

Maximise capturing, correlating, and managing plant data in your data storage infrastructure.

IDX Nexus, an evolving modular real-time data communication middleware framework, is highly suited to legacy system integration, IIoT and asset monitoring applications. A global mining company, based in South Africa, integrated the IDX Nexus Historian within its process optimisation toolkit to provide high-resolution, easily accessible storage of plant control system data. The mining toolkit comprises a suite of tools that support the mining company's control engineers in designing, deploying, and supporting its process control strategy at sites worldwide across the mining group.

The customer

In a drive to increase operational efficiency, low energy consumption and improved product quality that began approximately twenty-five years ago, a global mining company that specialises in extracting precious metals and minerals, began to employ Advanced Process Control (APC) techniques in conjunction with standard Proportional, Integral, Derivative (PID) to improve the control and thus efficiency and throughput of various stages of the platinum processing and extraction process.

The challenge: need for simple but efficient data logging

To ensure the effectiveness and proper optimisation of the APC techniques, a detailed historical record of, a historical account of the real-time data flowing through the process optimisation toolkit layer was required. Although the mining company used plant historians for its toolkit purposes, the data required was of higher resolution and shorter historical duration compared to the data fed into the plant historians.

Thus, a localised historian solution was sought that could provide sufficient performance for the required quantity of data to be stored while being based on technologies and tools that were readily understood and accessible to their control engineers.

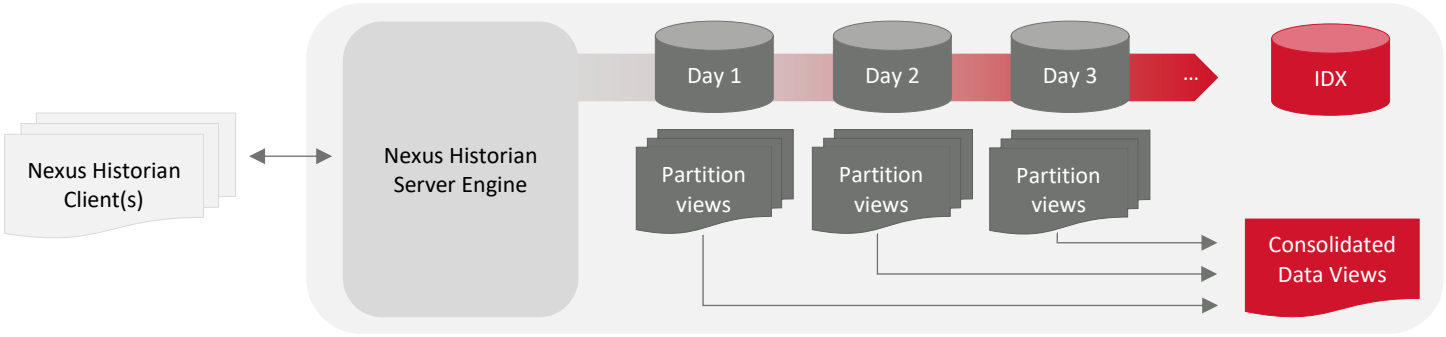
Why was the IDX solution selected?

The Nexus Historian was perfectly aligned with the envisioned requirements of the mining toolkits. While it does not carry the feature set of conventional enterprise-class historians, it is lightweight, simple to set up, and focuses on the task of storage and maintenance of time-series data. It uses the Microsoft Structured Query Language (SQL) Server as the storage database and can be queried by a very wide set of applications using standard SQL or Object Linking and Embedding Database (OLEDB) database connectors using the Transact-SQL (TSQL) query language that is widely understood and adopted.

It supports the storage of analogue (floating point), digital (integer, Boolean), string, and JavaScript Object Notation (JSON) data types. The Historian, however, is not a simple flat SQL table logger, which, while initially straightforward, tends to deliver increasingly poor performance as the table size grows and often poses challenges with data lifecycle management. The Nexus Historian is time-series data optimised and uses custom data partitioning techniques to enable large quantities of data to be stored using all versions of SQL Server, including the free SQL Express edition. It does this using some key configuration and operating principles.

The IDX Historian requires one important initial configuration decision to be made before the Historian can be used - what duration of data will be stored in a single database, in other words, what time frame or period will one database encompass - known as the Data Period. The Historian defines three possible Data Periods, namely Day, Week, or Month. This setting directly translates into the time ranges of data stored in a single SQL database, for example, if the Period is set to Day, each historian database contains one day's worth of data.

The Historian manages, writes and reads between the databases depending on the data timestamps such that the spread of data between different databases is not apparent to the user. For the user to obtain historical data, a query is made using the Nexus Historian Client, IDX database stored procedures, or views and not on the individual data storage databases. There is no limit to the number of Period databases beyond the limits imposed by host resources.



Nexus Historian Server/Client overview

Nexus Historian Server/Client Overview

Data is added to the Nexus Historian Server through the Client, which is directly integrated into the Nexus real-time data engine used by the process optimisation toolkit but is also available as a standalone .NET and COM API library.

The Client implements full data buffering such that if the Historian Server is not available to receive data for any reason, all client-recorded data is buffered locally until the server is restored. It also implements various data compression techniques, including the Swinging Door algorithm, to allow for reduced storage requirements without data change resolution loss.

For the toolkit's purpose, a historian period setting of Week was proven to be optimal, and up to a year's worth of data was mounted in the historian for analytical and reporting purposes. The Historian automatically performs backups and removes older periods where required.

Conclusion

The Nexus Historian provides a crucial data storage function for the mining company's toolkit use case. Its local, lightweight yet performant nature and that the data is accessible using common SQL-ready tools make it the ideal local historical data store for APC data.