

*Handy Generation of Java Domain Models from Databases*

Hazelcast Auto DB Integration is a time-saving tool for companies working with databases. It streamlines the development of Hazelcast applications by generating a Java domain model representation (POJOs and more) of the database, allowing companies to be productive with Hazelcast in no time.

## THE PROBLEM

If you write Java POJOs that do not change over time and no other applications are using the IMDG, then the process of populating a Hazelcast IMDG is straight forward.

However, if there are existing databases, if several applications share the IMDG, or if the data model changes from time to time, then a significant amount of boilerplate code needs to be written and maintained. Unfortunately, this makes the project more expensive, error-prone, and prolongs time-to-market.

## AUTO GENERATED DOMAIN MODEL

To address these problems, the Hazelcast Auto DB Integration extracts metadata from an existing database and generates code that supports features of the underlying database, the Hazelcast client, and IMDG. It provides an automatically generated domain model that includes POJOs (Portable), SerializationFactories, ClassDefinitions, MapStores, MapLoaders, ClientConfiguration, Ingest, Index and more.

## ELIMINATING MANUAL MAINTENANCE

Whenever the database schema changes, the process can be re-run again, effectively eliminating all manual work and errors with regards to maintaining the domain model.

Since many enterprise projects involve hundreds of tables, automating the process of writing and maintaining a domain model will significantly shorten time-to-market. At the same time, manual errors, project risks, and maintenance costs are reduced.

## EASY DEPLOYMENT

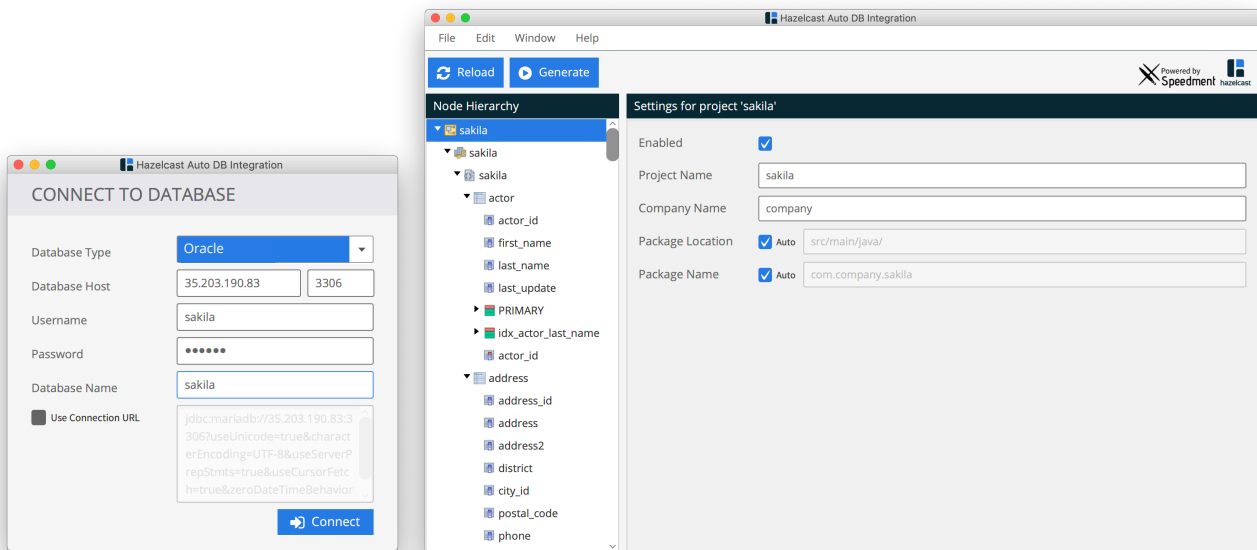
All common databases are supported and generated code can be run on-premise, in the cloud, Docker and Kubernetes. The generated domain model is compatible with Hazelcast Enterprise functions such as High-Density Memory Store, Hot Restart Store, Management Center etc.

Nodes in the IMDG do not need to have the generated domain model on their classpath. This simplifies setup, configuration and scaling of nodes in the IMDG and allows applications to be upgraded freely without disturbing other applications running on the same grid.

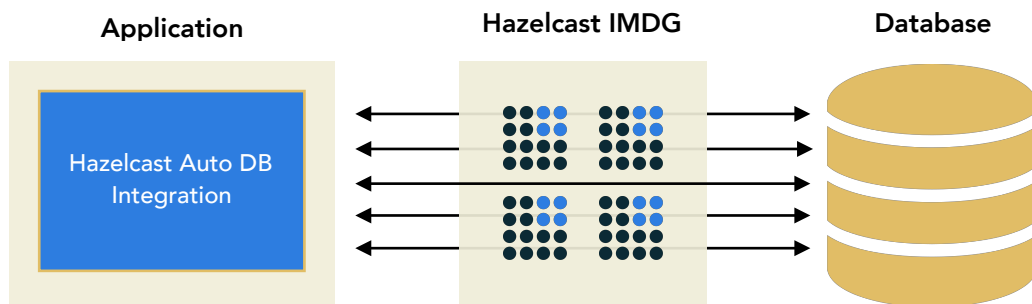
## KEY POINTS

- Automates the process of writing and maintaining a Java domain model of hundreds or even thousands of tables.
- Performs automatic schema migration.
- Keeps data safely in the database using write-through or write-behind operations.
- Provides initial data ingest support.
- Reduces development time and manual errors which yields faster time-to-market.
- Supports Oracle, SQL Server, AS400, DB2, MySQL, MariaDB, PostgreSQL, and SQLite.

**“Hazelcast Auto DB Integration removes a lot of work and brings companies to productivity with Hazelcast much faster”**



Once connected to the database, the tool visualizes the domain model in a tree structure and allows easy configuration. For applications which do not require access to the entire database, irrelevant tables or columns can simply be ignored by the code generator.



Reads and writes are then performed through or behind the Hazelcast IMDG while the data is kept safely in the database. Hazelcast Auto DB Integration also provides initial data ingest support.

## CONCLUSION

Hazelcast Auto DB Integration is a handy, time-saving tool for companies working with databases. It removes a lot of work and potential for errors, which can bring companies to productivity with Hazelcast faster.

For more information, please visit [hazelcast.com/product-features/database-integration/](https://hazelcast.com/product-features/database-integration/)

To download 30-day Trial of Hazelcast Auto DB Integration, please visit [speedment.com/hazelcast-initializer/](https://speedment.com/hazelcast-initializer/)

