Demystifying Famous Errors in Informix and Their Resolutions

1. **Error -23101:** Could not load module %INFORMIXDIR%/lib/esql/checkapi

**Resolution:** This error indicates a problem with the Informix ESQL/C library. To resolve it, perform the following steps:

* + Ensure that the %INFORMIXDIR% environment variable is set correctly and points to the Informix installation directory.
  + Verify that the ESQL/C library files are present in the specified directory.
  + Check the permissions of the ESQL/C library files and ensure they are accessible by the user running the application.

1. **Error -951:** Cannot create a trusted connection to the database server

**Resolution:** This error occurs when there is a mismatch in authentication settings between the Informix server and the client. Follow these steps to resolve it:

* + Ensure that the authentication method used by the Informix server matches the one configured in the client application.
  + Verify that the necessary user credentials are provided correctly in the connection string or configuration file.
  + If using Kerberos authentication, verify the Kerberos configuration and keytab files.

1. **Error -1213:** Deadlock detected

Resolution: Deadlocks occur when multiple transactions are waiting for resources held by each other, resulting in a circular dependency. To resolve a deadlock in Informix, consider the following steps:

* + Identify the transactions involved in the deadlock by examining the error message and logs.
  + Analyze the database schema and transaction logic to identify potential areas causing the deadlock.
  + Optimize your application code or SQL queries to minimize transaction duration and reduce the chances of deadlock.
  + Implement locking strategies, such as using explicit locks or adjusting isolation levels, to prevent conflicts and improve concurrency.

1. **Error -979:** Cannot modify table or view used in subquery

Resolution: This error occurs when attempting to modify a table or view that is being referenced in a subquery. To resolve it, consider the following:

* + Rewrite the query to avoid modifying the table or view directly in the subquery. Use temporary tables or intermediate steps, if necessary.
  + Ensure that the table or view being modified is not part of any subquery within the statement.

1. **Error -674:** Transaction is rolled back when accessing the database

Resolution: This error typically indicates a problem with transaction management. Follow these steps to resolve it:

* + Check for any previous errors or exceptions that might have caused the transaction to be rolled back.
  + Verify that the necessary BEGIN WORK and COMMIT/ROLLBACK statements are properly used in your application code.
  + Ensure that the transaction isolation level is set appropriately to prevent conflicts and unwanted rollbacks.

Understanding and resolving famous errors in Informix databases is crucial for maintaining a stable and efficient database environment. By familiarizing yourself with these errors and their resolutions, you can troubleshoot issues more effectively and ensure the smooth operation of your Informix applications.