Xaraya
Request for Comments: 0045
Category: Best Current Practice

M.R. van der Boom
Xaraya Development Group
April 2003

Multi database considerations

Status of this Memo

This document specifies a Xaraya Best Current Practices for the Xaraya Community, and requests discussion and suggestions for improvements. Distribution of this memo is unlimited.

Copyright Notice

Abstract

This RFC documents the considerations we take into account for supporting multiple databases. Each database has its own peculiarities and specific requirements. The document will be helpful to developers to ensure the code runs with all supported databases.

Table of Contents

1. Introduction

For each database we support there is a section in this document containing information which needs to be taken into account writing portable code for all supported databases.

Also some dedicated sections will be written for special situations.

2. Overall consideration

3. MySQL

4. PostgreSQL

5. Oracle

6. MS SQL Server

1. IDENTITY

IDENTITY fields cannot be used - use the normal Xaraya GenID() approach (cfr. ADODB xarmssql driver).

2. GROUP BY Fields

You cannot use column aliases for expressions in GROUP BY fields, but you can use that same expresssion as a GROUP BY field.

3. Data Types

Numeric values cannot be inserted directly in varchar/text fields. Use bind variables and type casting (cfr. xarVar__SetVarByAlias and modules_init()).

4. Binary Data (BLOB)

You need to use UpdateBlob() to insert data into a binary field (cfr. xarSession__phpWrite). Try to avoid binary fields and use text fields where possible.

5. Text Data (TEXT)

You may need to CAST text fields to VARCHAR(8000) [or smaller] when mixing data types (cfr. Dynamic_VariableTable_DataStore). No cross-database recommendations yet.

6. Empty Strings

From the Transact-SQL user guide : The empty string is interpreted as a single space in all char, varchar, nchar, nvarchar, and text concatenation, and in varchar insert and assignment statements.

From the SQL Server 2000 help : Interpretation of an empty string is controlled by the compatibility level, which is set with the sp_dbcmptlevel system stored procedure. If the compatibility level is 65 or lower, SQL Server interprets empty strings as single spaces. If the compatibility level is 70 or 80, SQL Server interprets empty strings as empty strings. For more information, see sp_dbcmptlevel. Expect differences in interpretation of empty strings compared to earlier versions of SQL Server.

In short, you have no idea what will be stored or what you'll get back. The good news is that SELECT ... FROM ... WHERE myfield = '' will work either way. The bad news is that if you try to check for empty($myfield) in PHP afterwards, you may get a difference since a single space is ... not empty.

Solution : the ADODB mssql driver was adapted to automatically replace single spaces by empty strings in results, so that you can forget about this problem...

7. Data Binding

Data binding concerns the linking of data in PHP to database queries. Binding can work in either direction - passing data from PHP variables to queries, and passing query results direct to PHP variables. For maximum portability, Xaraya only supports the passing of data from PHP into queries.

8. Revision history

2003-04-18: MrB: created

2003-05-05: Richard Cave: First stab at database requirements.

2003-05-14: Richard Cave: Added MySQL multi-table DELETE.

2003-06-12: Richard Cave: Added PostgreSQL SELECT DISTINCT with ORDER BY and updated index names.

2003-02-21: Jason Judge: Added section on data binding.

2005-07-02: Mike's Pub: Added section on MS SQL Server and comments about Binary Data/Text Data/HAVING