Table 13-7 Dilletelices between 1900 and visit	Table 19-2	Differences	Between	JSON and XML
--	-------------------	-------------	---------	--------------

Feature	JSON	XML
Useful for simple, structured data	Yes	Yes, but also supports semi- structured data and complex structured data
Useful for mixed content	No	Yes
Lacks attributes, namespaces, inheritance, and substitution	Yes	No
Places importance on ordering	No	Yes
Primarily intended for documents rather than data	No	Yes
Includes a date data type	No	Yes



Oracle Database JSON Developer's Guide for a more comprehensive comparison of XML and JSON

Native Database Support for JSON

JSON is widely stored in noSQL databases that lack relational database features. In contrast, Oracle Database supports JSON natively with features such as transactions, indexing, declarative querying, and views.

51

You can access JSON data stored in the database the same way you access other database data, including using OCI, .NET, and JDBC. Unlike XML data, which is stored using SQL data type XMLType, JSON data is stored using VARCHAR2, BLOB, or CLOB. By using Oracle SQL, you can perform the operations such as the following on JSON data:

- Join JSON data with non-JSON relational data.
- Generate JSON document from relational data using SQL functions json_object and json_array.
- Project JSON data into a relational format by using the SQL function json_table.
- Create a check constraint with is_json to enforce JSON data in a column. The
 database uses the check constraint to confirm that the column is JSON for JSONspecific operations such as simplified syntax.
- Manipulate JSON documents as PL/SQL objects.
- Use SQL functions json_query and json_value to accept an Oracle JSON path expression as an argument and match it against the target JSON data.
- Index JSON data.
- Query JSON data stored in an external table.
- Replicate tables with columns containing JSON data using Oracle GoldenGate.

Textual JSON data always uses the Unicode character set, either UTF8 or UTF16. Oracle Database uses UTF8 internally when it parses and queries JSON data. If the

3.5.1 About Client Result Cache

51

Applications that use Oracle Database drivers and adapters built on OCI libraries—including C, C++, Java (JDBC-OCI), PHP, Python, Ruby, and Perl—can use client result cache to improve response times of repetitive queries.

Client result cache enables client-side caching of SQL query (SELECT statement) result sets in client memory. Because retrieving results from a client process is faster than calling the database and rerunning the query, frequently run queries perform significantly faster when their results are cached. Client result cache also reduces the server CPU time that would have been used to process the query, thereby improving server scalability.

OCI statements from multiple sessions can match the same cached result set in the OCI process memory if they have similar schemas, SQL text, bind values, and session settings. If not, the query execution is directed to the server.

Client result cache is transparent to applications, and its cache of result set data is kept consistent with session or database changes that affect its result set data.

Applications that use client result cache benefit from faster performance for queries that have cache hits. These applications use the cached result sets on clients or middle tiers.

Client result cache works with OCI features such as the OCI session pool, the OCI connection pool, DRCP, and OCI transparent application failover (TAF).

When using client result cache, you must also enable OCI statement caching or cache statements at the application level.

See Also:

- Oracle Call Interface Programmer's Guide for information about statement caching
- Oracle Database JDBC Developer's Guide for information about JDBC statement caching
- Oracle Call Interface Programmer's Guide
- Oracle C++ Call Interface Programmer's Guide
- Oracle Database JDBC Developer's Guide
- Oracle Database Performance Tuning Guide
- Oracle Database Concepts
- SQL hints and RESULT_CACHE for clauses of ALTER TABLE and CREATE TABLE
- RESULT CACHE MODE
- CLIENT_RESULT_CACHE_STAT\$ view

