

## eDominator EXPAND smERP *RDBMS vs OODBMS*



07 05 001 (07/05)

© 2006 by eDominator Systems Private Limited. All rights reserved. e-Dominator, eDominator.com, and other eDominator products and services mentioned herein as well as their respective logos are trademarks or registered trademarks of eDominator Systems Private Limited in India. All other product and service names mentioned are the trademarks of their respective companies. Data contained in this document serves informational purposes only. Product specifications may vary. Printed on environmentally friendly paper. These materials are subject to change without notice. These materials are provided by eDominator Systems Private Limited and its affiliated companies for informational purposes only, without representation or warranty of any kind, and eDominator shall not be liable for errors or omissions with respect to the materials. Reproduction, adaptation or translation is prohibited without the express written permission of eDominator Systems Private Limited. eDominator shall not be liable for technical or editorial errors or omissions contained herein.

“More than 90% of Enterprise Applications across the globe run on RDBMS”

RDBMS	OODBMS
<p>RDBMS has more than 90% of the Database market.</p> <p>© 2002 Jaelson Castro and John Mylopoulos</p>	<p>OODBMS has less than 5% of the Database market.</p> <p>© 2002 Jaelson Castro and John Mylopoulos</p>
<p>RDBMS can handle <math>10^{10}</math> records.</p> <p>© 2002 Jaelson Castro and John Mylopoulos</p>	<p>OODBMS can handle only up to <math>10^7</math>.</p> <p>© 2002 Jaelson Castro and John Mylopoulos</p>
<p>More than 90% of the world’s enterprise applications use RDBMS technology from leading vendors like Microsoft, Oracle &amp; IBM.</p> <p>Source: Gartner Dataquest 2004</p>	<p>OODBMS’s main disadvantage is that their technology is immature and they are only used in applications such as CAD.</p> <p>© 2002 Jaelson Castro and John Mylopoulos</p>
<p>In an RDBMS modifying the database schema either by creating, updating or deleting tables is typically independent of the actual application.</p>	<p>In an OODBMS based application modifying the schema by creating, updating or modifying a persistent class typically means that changes have to be made to the other classes in the application that interact with instances of that class. This typically means that all schema changes in an OODBMS will involve a system wide recompile. Also updating all the instance objects within the database can take an extended period of time depending on the size of the database.</p>
<p>This is not the case in RDBMS as the database structure is independent of the programming language used.</p>	<p>An OODBMS is typically tied to a specific language via a specific API. This means that data in an OODBMS is typically only accessible from a specific language using a specific AP.</p>

<p>In an RDBMS, the relational nature of the data allows one to construct ad-hoc queries where new tables are created from joining existing tables then querying them.</p>	<p>Lac of Ad-Hoc queries. Since it is currently not possible to duplicate the semantics of joining two tables by "joining" two classes then there is a loss of flexibility with an OODBMS. Thus the queries that can be performed on the data in an OODBMS is highly dependent on the design of the system.</p>
<p>RDBMS can handle large data which can run into terabytes of information.</p>	<p>OODBMS is incapable of handling large data.</p>