Bringing R to the Enterprise

A Familiar R Environment with Enterprise-Caliber Performance, Scalability, and Security
**Disclaimer**

The following is intended to outline our general product direction. It is intended for information purposes only, and may not be incorporated into any contract. It is not a commitment to deliver any material, code, or functionality, and should not be relied upon in making purchasing decisions. The development, release, and timing of any features or functionality described for Oracle’s products remains at the sole discretion of Oracle.
Introduction: Beyond the Laptop ......................................................... 4
Run R Code in the Database ........................................................... 5
Enjoy In-Database Support for CRAN Packages ............................ 5
Operationalize R Analytics .............................................................. 6
Run R with Hadoop ......................................................................... 6
Big Data Use Cases with Oracle Database and Hadoop ............... 6
  Use Case 1: Analyzing Credit Risk ................................................ 6
  Use Case 2: Detecting Fraud .......................................................... 7
  Use Case 3: Preventing Customer Churn ....................................... 8
Conclusion: Analytics for the Enterprise ........................................... 8
Introduction: Beyond the Laptop

Formulated in 1994 as an alternative to proprietary statistical environments, R is an integrated suite of software facilities for data manipulation, calculation and graphical display. This open source scripting language has become an important part of the IT arsenal for analysts and data scientists conducting statistical analysis on big data. There are now about 2 million R users worldwide who utilize thousands of open sources packages within the R ecosystem to enhance productivity in such domains as bioinformatics, spatial statistics, financial market analysis, and linear/non-linear modeling.

While many developers and analysts run R programs on their personal computers, sometimes they need to do advanced computations on large amounts of data in a short period of time. To assist these developers, Oracle has created a broad set of options for conducting statistical and graphical analyses on data stored in Hadoop or Oracle Database, bringing enterprise caliber capabilities to projects that require high levels of security, scalability and performance.

Oracle Advanced Analytics and Oracle R Connector for Hadoop combine the advantages of R with the power and scalability of Oracle Database and Hadoop. R programs and libraries can be used in conjunction with these database assets to process large amounts of data in a secure environment. Customers can build statistical models and execute them against local data stores as well as run R commands and scripts against data stored in a secure corporate database.

These capabilities are especially important for many of today’s big data projects. Data analysts can obtain controlled access to data in Oracle Database, accelerating productivity while enforcing IT security policies. Oracle’s integrated approach simplifies data analysis, eliminates data movement, and shortens the time it takes to transform raw data into actionable information.

Oracle Advanced Analytics links R to Oracle Database. Oracle Advanced Analytics, an option of Oracle Database Enterprise Edition, empowers data and business analysts to extract knowledge, discover new insights and make predictions—working directly with large data volumes in the Oracle Database. Oracle Advanced Analytics offers a combination of powerful in-database algorithms and open source R algorithms, accessible via SQL and R languages, and provides a range of GUI and IDE options targeting the spectrum from business users to data scientists.

In addition, Oracle R Connector for Hadoop, one of the Oracle Big Data Connectors, links R to Hadoop. If you have data spread out in Hadoop Distributed File System (HDFS) clusters, you can use tools like Hive to lend structure to the data and create a table-like environment. Analysts can then run queries against the HDFS files and execute the latest R packages inside the database.

Oracle Advanced Analytics can be used in conjunction with a multi-threaded version of open source R with enhanced mathematical libraries. Oracle R Distribution can be readily enhanced with high performance libraries such as Intel’s MK for enhanced linear algebra and matrix processing. Oracle R Distribution is supported by Oracle. It uses in-database statistical techniques to optimize data transport and can leverage data-parallelism constructs within the database for enhanced scalability and performance.

In the remainder of this paper, we will demonstrate how Oracle enhances open source R by enabling developers to:

• Transparently analyze and manipulate data in Oracle Database or Hadoop
• Execute R scripts through the database with data and task parallelism
• Use in-database SQL based Oracle Data Mining algorithms seamlessly through R
• Score R models in the database
• Automatically integrate R scripts into SQL statements
• Integrate R into the IT software stack

By integrating R with both Oracle’s core database and infrastructure offerings and Hadoop, your organization can realize the best of both worlds: you will obtain a familiar yet powerful statistical environment along with vastly improved scalability, security and performance.

Run R Code in the Database

Oracle developed a set of packages on top of the R engine that allows R computations to be executed from within Oracle Database. This “transparency layer” makes Oracle tables and views accessible to the R environment as if they were native R objects, allowing developers to execute a wide range of R functionality. Data analysts can use their favorite R GUI for exploratory data analysis—without having to know SQL. R allows statisticians and data scientists to access a wide range of analytical capabilities in a natural statistical language – R. This lets them focus on data analysis challenges rather than data access challenges.

The transparency layer allows R developers to use familiar environments, languages and tools. Under the covers they execute on the database, in parallel, with a rich library of statistical functionality. R users can execute these complex computations within the database by resorting to their standard R development skills and tools. They can build, evaluate, share, and deploy predictive analytics methodologies, while also utilizing high-performance Oracle-specific data mining algorithms.

Developers can easily scale their analytic projects as data volume increases by bringing the algorithms to where the data resides. For example, they can utilize Oracle Data Mining native SQL based algorithms like decision trees, support vector machines, clustering from within the same transparency process, deploy neural networks to make it extremely scalable, and utilize linear models with stepwise regression. To enhance its R implementation, Oracle has implemented data mining algorithms that can build and apply models while the data remains in the database. They can analyze data and make predictions even faster when they run in Oracle Exadata, since this processing takes place at the storage tier. This allows organizations to take advantage of the extreme performance of Oracle Engineered Systems for virtually limitless performance and scale. The benefits of the Oracle approach are clear:

• Work solely from within R for data preparation, analysis, and visualization
• Use the database as compute engine with query optimization and parallelism
• No need to manage flat file data, or wrestle with the associated complexity of storage, backup, recovery, and security
• Eliminate R memory constraints so you can handle big data requirements
• Embed R statements in SQL

Enjoy In-Database Support for CRAN Packages

Oracle’s Embedded R execution layer allows developers to execute thousands of specialized algorithms from the Comprehensive R Archive Network (CRAN) repository. They can write their own algorithms or download existing ones, then install these packages in the Embedded R server, on the same computer as the database. This architecture makes it easy to send and receive data to and from the database and feed it directly to their chosen algorithms.
By taking advantage of parallel feeds through indexing it is possible to run exotic algorithms. For example, you might divide a customer database by zip code and run multiple sessions in parallel to process groups of customers from many different zip codes simultaneously, all without leaving the R environment. R scripts that expose a wide variety of statistical techniques—some accessible through the transparency layer in Oracle Advanced Analytics and some in CRAN packages—can be built and stored in Oracle’s in-database R script repository.

- Create your own packages in R and execute them in the database
- Leverage CRAN open source packages
- Enable “lights-out” execution of R scripts via a SQL interface
- Speed up large jobs with parallel database execution
- Integrate results with BI dashboards and reports

Operationalize R Analytics

Oracle Advanced Analytics enables R developers to use the database to execute R scripts within SQL queries. This makes it easy to operationalize R scripts within a standard business intelligence environment. Any SQL query to Oracle Database can contain a call to an R script that is registered in the database R script repository. Using the script name, you can initiate a query to call that script and receive the results in a new table, chart or graph. For example, parameters controlling R models can be passed as run-time arguments to programmatically update BI dashboards and graphical reporting applications.

One telecommunications provider used Oracle Advanced Analytics to power some complex survey research. Analysts at this firm maintain analytic functions in Oracle Database, then filter the data and display the results through a parameterized BI dashboard. Both the database and the BI infrastructure are standard components of the architecture, further enhanced by their connection to the R scripts. These capabilities make R a more powerful language that can execute advanced statistical models directly.

Run R with Hadoop

Oracle R Connector for Hadoop (ORCH) is an R package that provides transparent access to Hadoop and data stored in HDFS. This Oracle Big Data Connector enables users to run R models efficiently against large volumes of data, as well as leverage MapReduce processes without having to leave the R environment. They can use R to analyze data stored in HDFS with Oracle supplied analytics, as well as utilize over 3,500 open source R packages. ORCH enables R scripts to run on data in Hive tables and files in HDFS by leveraging the transparency layer as defined in Oracle R Enterprise. Hadoop-based R programs can be deployed on a Hadoop cluster. Users don’t need to know about Hadoop internals, command line interfaces, or the IT infrastructure to create, run, and store these R scripts.

Big Data Use Cases with Oracle Database and Hadoop

Oracle’s big data technologies are designed to easily move data between Hadoop, R, and Oracle Database. Now analysts can access data stored in Oracle or Hadoop and can code MapReduce algorithms in R without having to resort to Java. As we will see in the examples that follow, this flexible architecture enables organizations to easily analyze large tables and large data sets as they solve today’s pressing big data challenges. In addition to SQL, R is now a good option for enterprise analytics to solve today’s pressing big data challenges.

Use Case 1: Analyzing Credit Risk
Banks continually offer new services to their customers, but the terms of these offers vary based on each customer’s credit status. Do they pay the minimum amount due on credit balances, or more? Are their payments ever late? How much of their credit lines do they use and how many other credit lines do they have? What is the overall debt-to-income ratio?

All of these variables influence policies about how much credit to award to each customer, and what type of terms to offer them. A bureau like Equifax or Transunion examines an individual’s overall credit history. But banks can examine a much more detailed set of records about their customers—down to the level of every discrete transaction. They need big data analytics to get down to this level of precision with this volume of data.

For example, one Oracle customer in Brazil is running multiple neural network algorithms against hundreds of millions of records to examine thousands of attributes about each of its customers. Previously the bank had trouble crunching this massive volume of data to generate meaningful statistics. They solved this problem by running a specialized algorithm using Oracle R Connector for Hadoop to analyze this data in parallel on the same cluster that is running the Hadoop file system, Hive, and other tools. The Oracle R Connector for Hadoop enables analysts to execute R analysis, statistics and models on tables stored in the bank’s large Hadoop file systems. They can now run complex statistical algorithms against these files systems and Hive tables. They can also use specialized Mahout algorithms to perform unique analyses.

The Oracle algorithms are transparent to R users and accessed with a couple of lines of code. Behind the scenes, ORCH provides the interface for executing MapReduce jobs in parallel on multiple processors throughout the bank’s cluster. Analysts can create these MapReduce algorithms in R and store them in Hadoop as well as easily surface these models for review, plotting and analysis—and then push them to the database—without having to utilize Java.

Use Case 2: Detecting Fraud

Another popular use case involves detecting fraud by analyzing financial transactions. Banks, retailers, credit card companies, telecommunications firms and many other large organizations wrestle with this issue. When scoring fraud you typically study transactions as they occur within customer accounts. (Scoring refers to predicting outcomes using a data-mining model.)

Once you understand the modus operandi – the normal mode of operations – you can then recognize unusual patterns and suspicious transactions. For example, if you normally shop in Los Angeles and there is a sudden series of transactions in Rome this would indicate a high likelihood of fraud. Or would it? If you are somebody who travels a lot, is a surge of activity in Rome an anomaly or a regular pattern? By capturing all previous transactions and studying these patterns you can develop a model that reflects normal behavior.

R has the ideal algorithms and environment for creating a scoring model that can analyze these transactions. But because it was conceived as a single user tool it is not multi-threaded. The client and server components are bundled together as a single executable, much like Excel. Thus with open source R the scoring model is limited by the memory and processing power of the machine on which it runs. It can’t even leverage the CPU capacity of a multi-processor laptop without special packages and programming—or without a special implementation.

Oracle Advanced Analytics can handle the massive computational requirements associated with analyzing customer-purchasing patterns using the R language to define scripts that get pushed down and run in the database. Organizations can leverage Oracle Exadata, Oracle Big Data Appliance, and Oracle Exalytics to scale the effort, and integrate Oracle Business Intelligence Enterprise Edition to display the results. R developers can take the fraud model out of HDFS and put it in Oracle Database where it can rapidly predict behavior at the transactional level and be part of an enterprise application for real-time predictions.
Real-time analysis is important in fraud prevention. It’s one thing to identify a fraudulent transaction that happened eight hours ago (the length of time it might take to stage data into a laptop and run a detailed analysis of yesterday’s activities). It’s clearly much more valuable to score that transaction against a model in real-time, with the potential to block the transaction of flag it for further scrutiny.

Use Case 3: Preventing Customer Churn

Customer churn is a major problem for many businesses, especially in highly competitive markets such as telecommunications. For example, as a mobile phone user, if you have problems with reception or experience too many dropped calls you might think about looking for another service provider. Your existing service provider is constantly analyzing your behavior to predict how likely you are to defect. They have a statistical model that says, “90% of our customers with similar issues and behavior have left us for another service provider.” They can apply that model to your data to create a score that reveals your likelihood of defecting. Your score relates you to millions of other customers with similar behavior.

Using Oracle Advanced Analytics you can run these R models while a customer is browsing a webpage or using a mobile app and then make on-the-spot recommendations based on current actions and real-time analytics against an operational data store or data warehouse.

Scoring can also be done offline. For example, you might want to predict which of your 100 million customers will respond to each of a dozen offers so you can identify which customers should be targeted with a special offer or ad campaign. With enough processing power and the right scoring model, analysts can provide insight not only into what the churn rate is but also the reasons behind the churn.

One telecommunications company used Oracle R Connector for Hadoop to make richer, more informed decisions by examining payment records, calling plans and service histories to detect similarities and trends within its customer databases. This connector permitted them to run batch jobs in parallel on a large Hadoop cluster.

Oracle provides options for executing these computations from the R environment on data resident in HDFS, Oracle Database, and local files.

Conclusion: Analytics for the Enterprise

Most organizations depend on databases to securely store information with rigorous, enterprise-level controls. Oracle has made R highly compatible with large-scale data mining, BI, and big data initiatives. Developers can utilize the familiar R environment in conjunction with Oracle Database, Hadoop, and business intelligence implementations as they apply massive scalability, reach and performance to big data problems. Analysts who are accustomed to working with file extracts can adopt a database-centric architecture, pushing data from their desktop R implementations to the database and using desktop models to process data that resides in a database.

R-to-SQL transparency improves user efficiency by allowing analysts to use R directly against data in an Oracle database. R users can leverage in-database SQL analytic and data mining functions and open-source R packages in combination with the database for task-parallel execution.

With Oracle Advanced Analytics, you can remain in your standard R environment. You can execute existing CRAN packages and other R assets that you have created, and embed R-based analytics into SQL statements to put scoring models into production and link them with your BI tools. For big data problems you can leverage the
scalability of Oracle R Distribution and Oracle R Connector for Hadoop with Oracle Big Data Appliance and Hadoop.

Customers that purchase Oracle Advanced Analytics, Oracle R Connector for Hadoop or Oracle Linux receive enterprise class support for Oracle R Distribution.

In summary, by extending R to work with Oracle Database and Hadoop you can bring your analyses to the data, rather than the other way around. By pushing R commands to Oracle Database via ORE and invoking R Hadoop jobs on Hadoop cluster nodes via ORCH, analysts can minimize data movement and decrease latency time from raw data to actionable information. Integrating these three popular environments provides a powerful, cost-effective solution for big data analytics.