



Microsoft R lab

Jan Kesters

jakester@microsoft.com



Agenda – Day 1

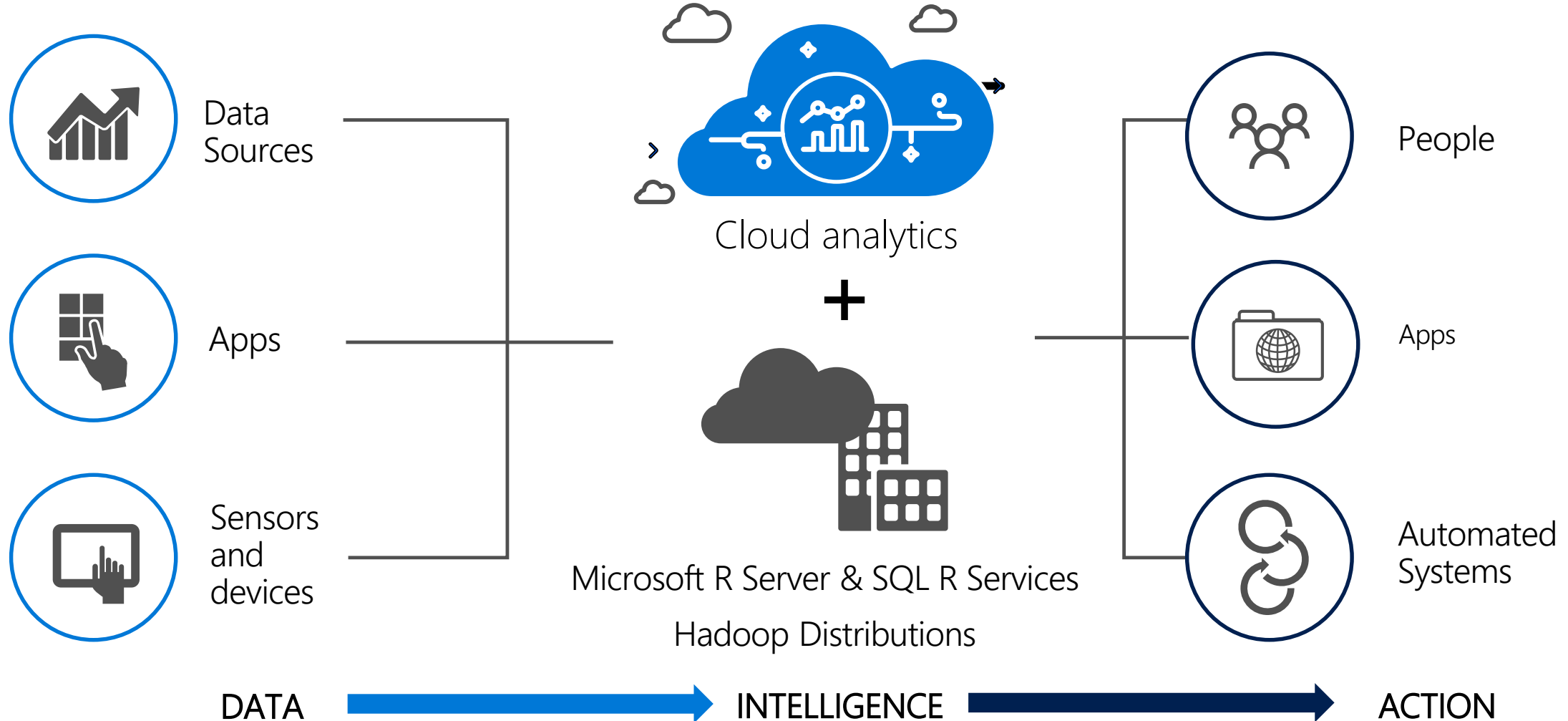
Who	When	What	How
All	09:30 – 09:45	Coffee, Introductions, Connectivity !	
Instructors	09:45 – 11:00	Microsoft R Server (MRS)	Presentation
You	11:00 – 12:00	Lab 01 : Introduction to Microsoft R Server	Lab
All	12:00 – 13:00	< LUNCH >	
You	13:00 – 14:30	Lab 02 : Data Cleansing & Management with MRS	Lab
You	14:30 – 15:30	Lab 03 : Building Predictive Models with MRS	Lab
All	15:30 – 15:45	< BREAK >	
You	15:45 – 17:00	Lab 04 : Free Lab with MRS	Lab
All	17:00 – 17:15	Wrap-Up : Questions and Answers	Discussion

Agenda – Day 2

Who	When	What	How
Instructors	09:30 – 11:00	Set-up Spark cluster R Deployment Options R Operationalization	Chalk & Talk
You		Lab 05 – Operationalizing R with Azure Machine Learning	Lab
You		Lab 08 – SQL Server R Services	Lab
All	12:00 – 13:00	< LUNCH >	
You		Lab 07 – Getting Started with MRS on HDInsights (Spark)	Lab
All	16:00 – 16:30	Wrap-up	Discussion

From data to intelligence to action

On Prem or in the Cloud



Microsoft Data Platform



SQL Server

- Everything built-in
- OLTP, analytical and MPP
- Continuous innovation
- Deep integration with Hadoop using T-SQL
- R Integration



Microsoft R Platform




- Enterprise-ready R
- Multi-threaded
- Massive Parallel Processing
- Reproducible R
- Fast path to industrialization
- Commercial support



Cortana Intelligence

- Scalable value-adding services
- Stand-alone or hybrid solutions
- Perceptual intelligence
- Business templates
- Industrialize in seconds
- Open platform (R, Spark, Python)

Comparing CRAN R vs Microsoft R

	 CRAN R	 Microsoft R Open	 Microsoft R Server
Speed of Analysis	Single threaded	Multi-threaded	Multi-threaded, parallel processing 1:N servers
Support	Community	Community	Community + Commercial
Analytic Breadth & Depth	8000+ innovative analytic packages	8000+ innovative analytic packages with a fixed CRAN repository	8000+ innovative analytic packages with a fixed CRAN repository + commercial parallel high-speed functions
Industrialization & Code Management	Custom	Custom	Deploy as web service Version control with Team Services
License	Open source	Open source mran.microsoft.com	Commercial license Supported release with indemnity

An R Server for Everyone

Microsoft R Server for Redhat Linux



Microsoft R Server for SUSE Linux



Microsoft R Server for Teradata



Microsoft R Server for Hadoop on Redhat



SQL Server R Services



Microsoft R Server for HDInsight with Spark

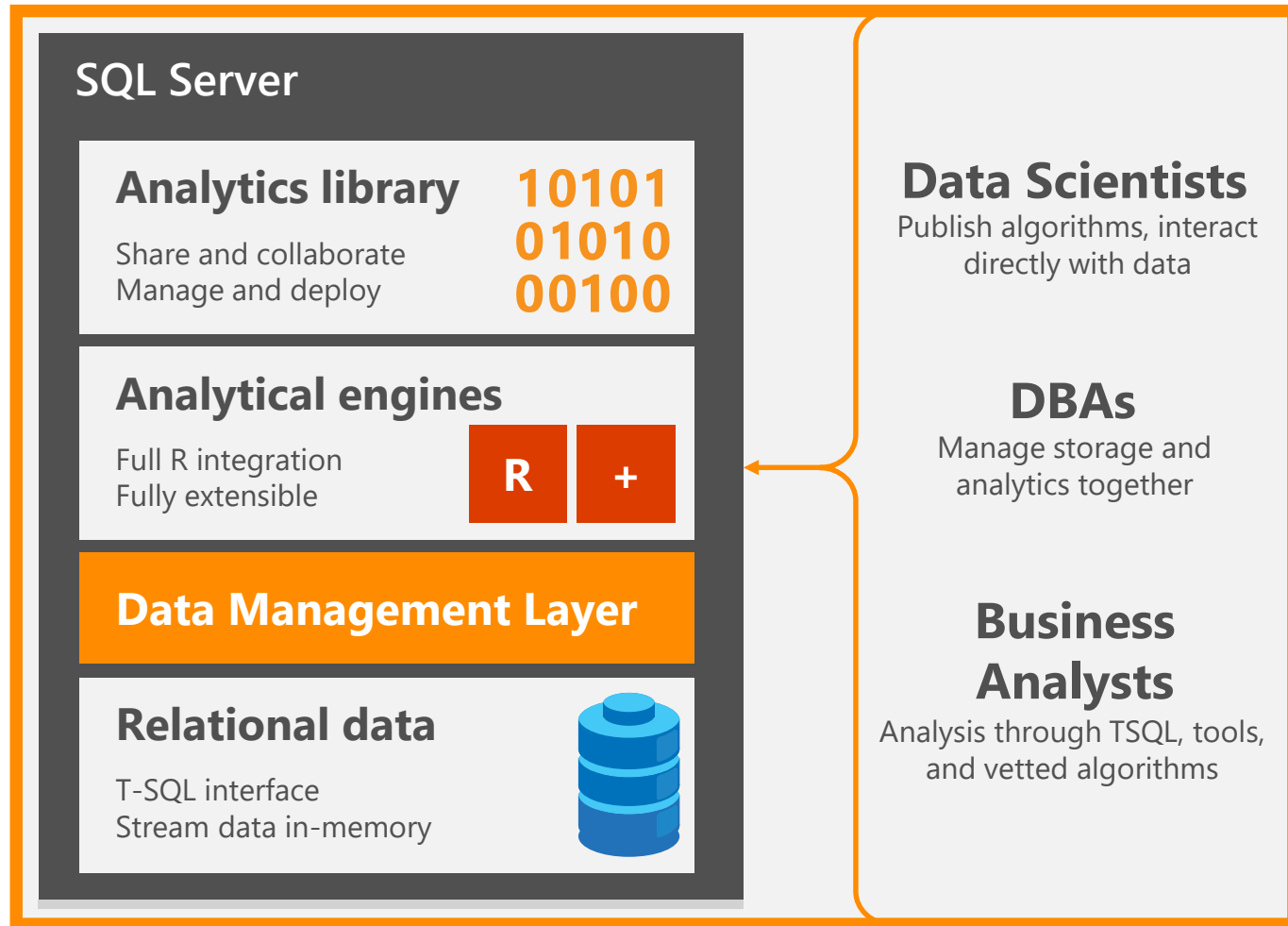


Introducing SQL Server 2016 R Services

Enterprise-scale data science—built-in







Simplicity & agility	Scalability & choice	Cost effectiveness
<p>Enterprise speed & performance</p> <p>Near-DB analytics</p> <p>Parallel threading & processing</p>	<p>Model on-premises, store in cloud—or vice versa</p> <p>Hybrid memory & disk scalability</p> <p>Not bound by memory-enabling limits of larger datasets</p>	<p>Included in SQL Server 2016</p> <p>Reuse and optimize existing R code</p> <p>Eliminate data movement across machines</p> <p>Write once, deploy anywhere</p>

Microsoft SQL Server R Services



- Working from the R IDE and execute R Scripts that runs in-Database
- Can manage, secure and govern resources of R Runtime Execution
- Execute the R Scripts, invoking T-SQL
- Run R Jobs

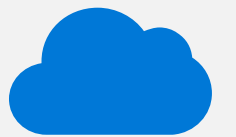
SQL Server & Power BI Everything Build-In

Future proof Hybrid Database	Most secure database 6 years in a row	Mission critical Database Technology	End-to-end Mobile BI on any device & Browser	In-database Advanced Analytics
<p>Backup to cloud Stretch Database </p> <p>AlwaysOn: DR in cloud Database as a Service</p> <p>Polybase </p>	<p>Advanced Threat Analytics SQL Server Auditing</p> <p>Row-Level Security </p> <p>Dynamic Data Masking </p> <p>Always Encrypted Transparent Data Encryption</p>	<p>In-Memory Technology for OLTP & DWH</p> <p>Row & Column Store</p> <p>Real-time Operational Analytics</p> <p>Industry Leading TCO</p>	<p>Online & Offline access Rich Visualization Power BI Integration</p> <p></p>	<p>Machine Learning</p> <p>Cost efficient Operationalize Scale & Performance Support </p> <p>R + in-memory at massive scale</p>

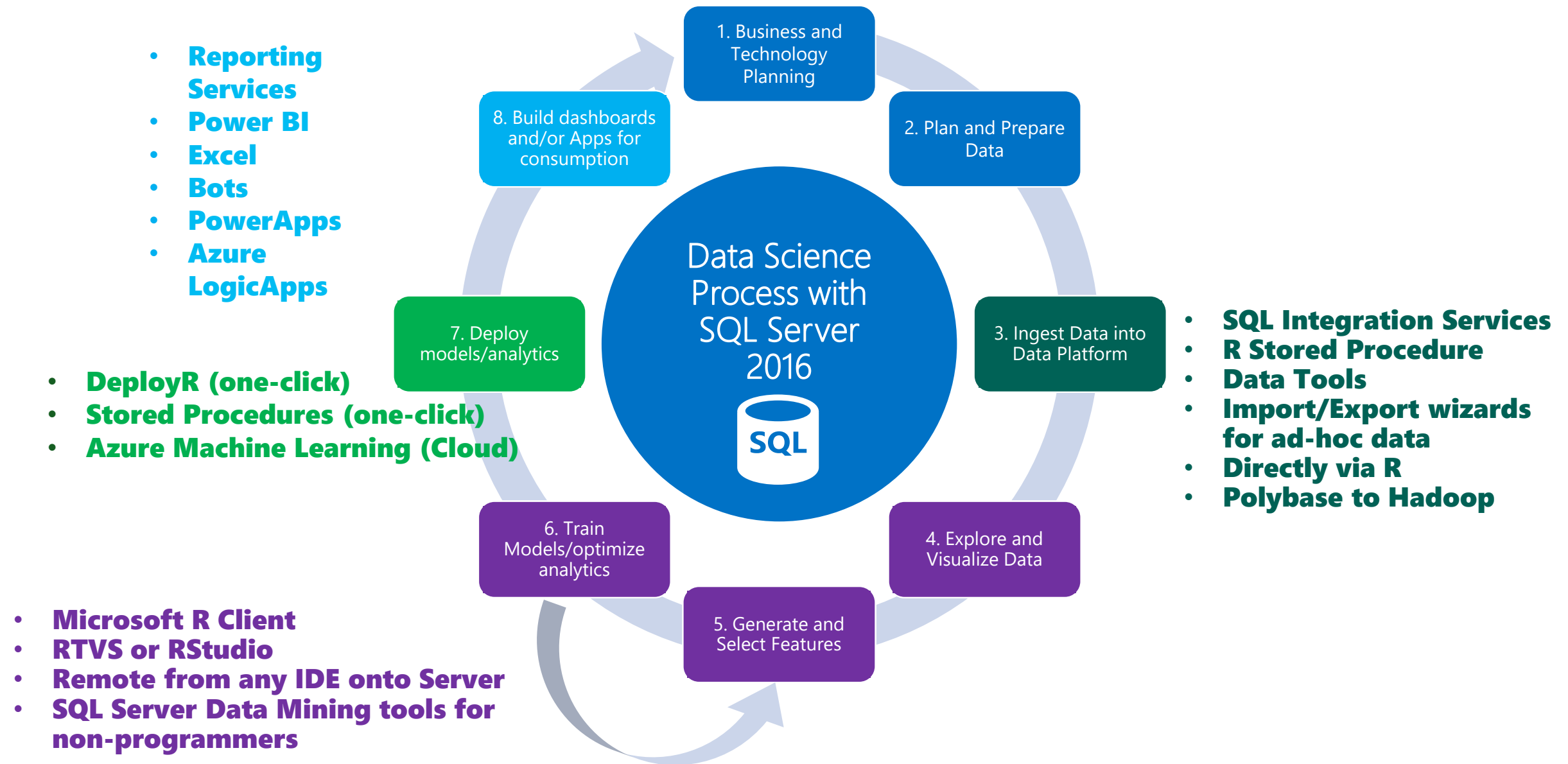
← In-memory across all workloads →



Consistent experience from on-premises to cloud



The Data Science Process with SQL Server 2016



Is your problem big enough for Hadoop ?

When to use Hadoop ?

- Conventional tools won't work on your data
- Your data is really big !
- Won't fit/process in your favourite database or file-system
- Data is really diverse !
- Semi-structured – JSON, XML, Logs, Images, Sounds
- You're a whiz at programming and sys-admin

When not to use Hadoop ?

- !(When to use Hadoop ?)
- You're in a hurry !



When and Where Should I Use Hadoop ?

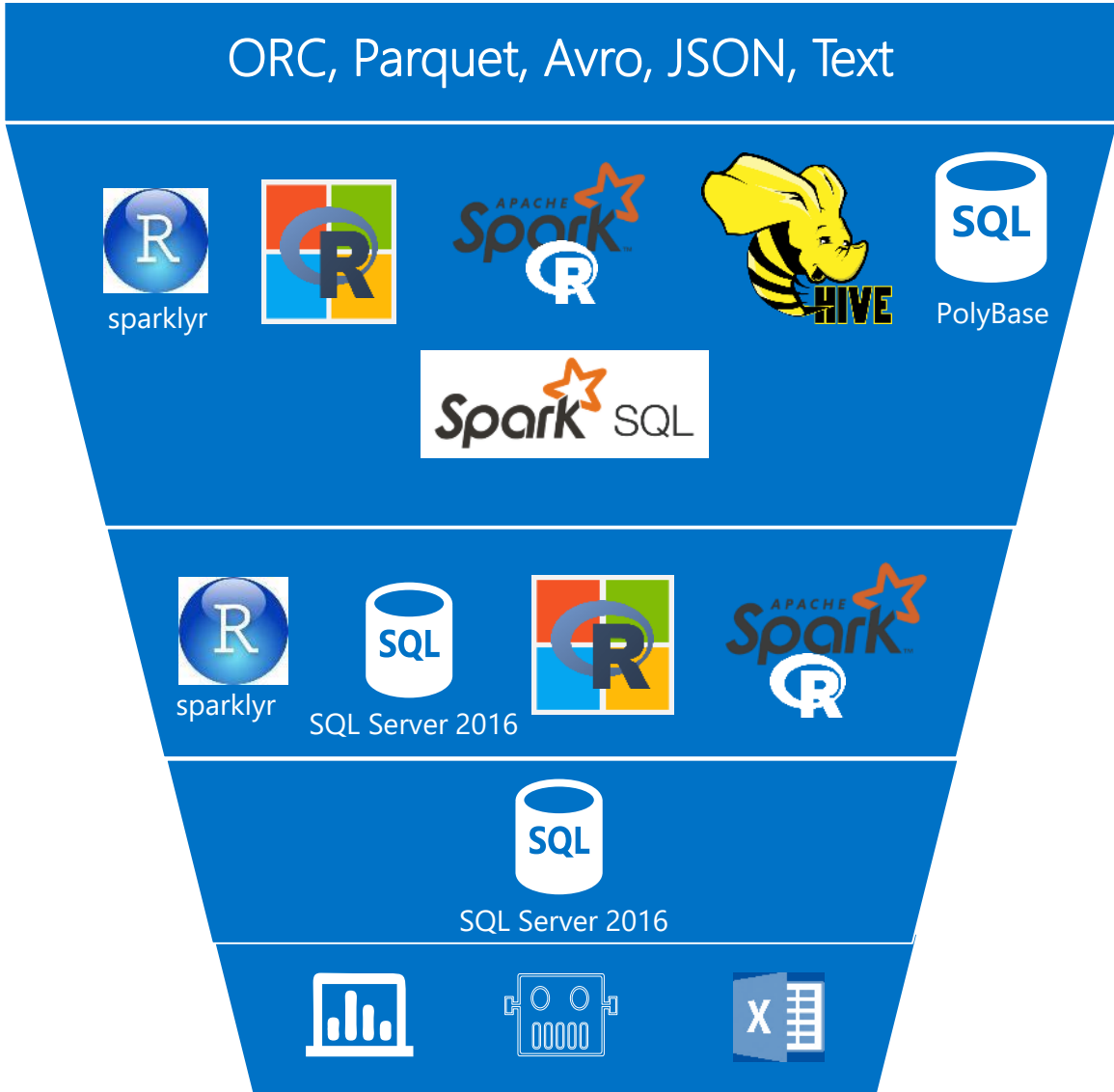
Store

Prepare

Model

Deploy

Consume



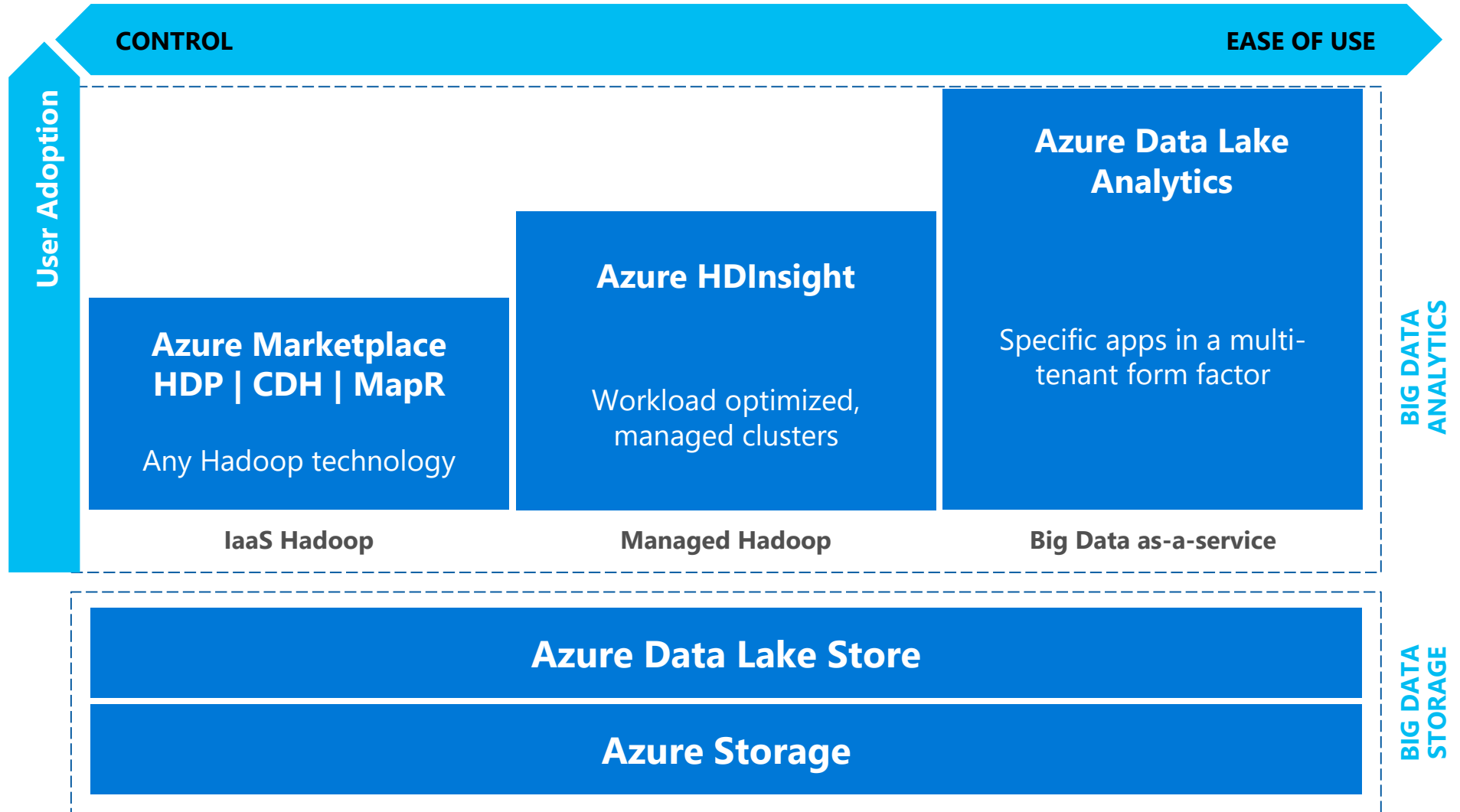
**Big Data
(>5TB)**

**Large Data
(<1TB)**

**Small Data
(MB)**

Choices of a Big Data Platform

Accelerate the pace of innovation through a state-of-the-art cloud platform

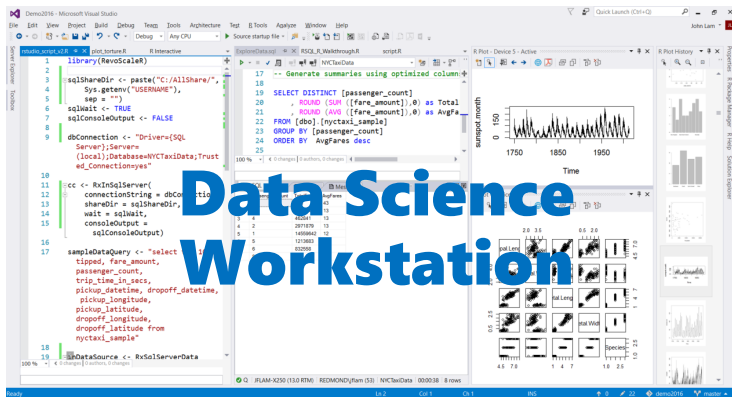


R in Hadoop: Options



Edge Node

DeployR Remote



Data Science Workstation

Head Node

Head Node

Data Node

Data Node

Data Node

Data Node

Data Node

Data Node

Data Node

Data Node

Data Node

Data Node

Data Node

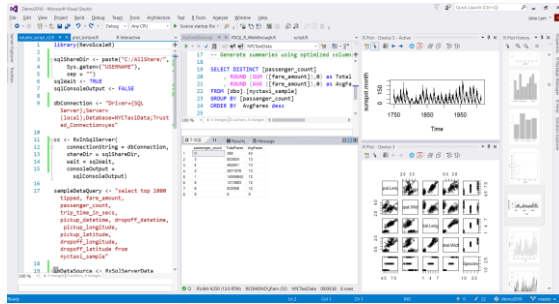
Data Node

High Level Architecture

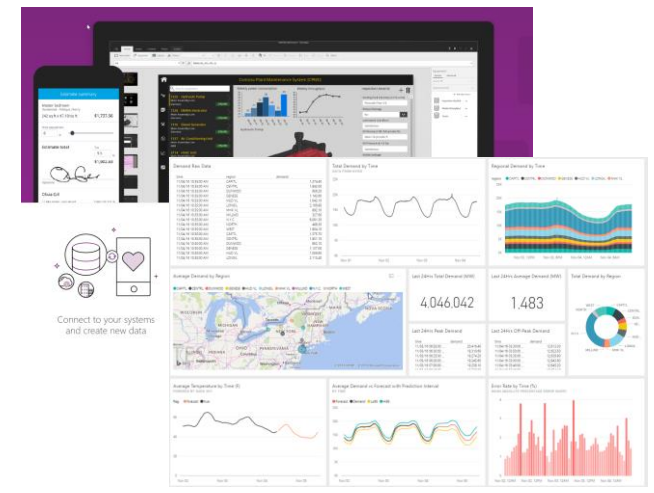
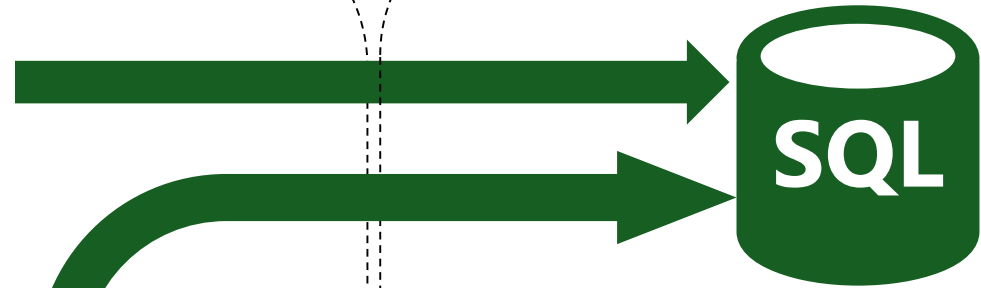
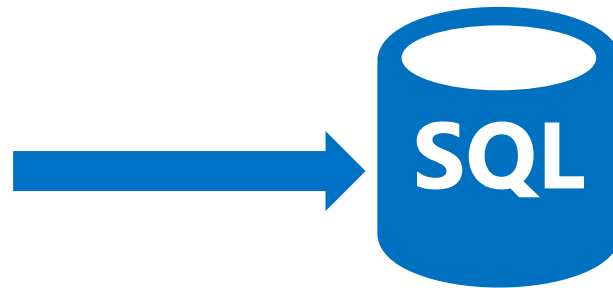
Once process has been tested in analytics environment, we promote to operational environment.

Analytics Environment

Operational



Data Scientists Workstations



From RStudio/RTVS remote to SQL and/or Hadoop Edge using DeployR R package

Continue to use local (workstation) for ad-hoc work

Productivity Tools: Microsoft R Client

The screenshot displays the Microsoft R Client interface. The top window shows R code for connecting to Hadoop and processing data. The bottom window shows R code for connecting to a SQL Server and querying data. The interface includes a console, a data viewer, and a plot area.

```
1 # Hadoop settings
2 -----
3
4 hadoopCluster <- RxHadoopMR(consoleOutput = TRUE)
5 hdfsFS <- RxHdfsFileSystem()
6
7 # Compute context options
8 -----
9 # 1. local / local
10 rxSetComputeContext("localpar")
11 mortDS <- RxKdfData("mortgages")
12
13 # 2. local / stream XDF data from HDFS
14 rxSetComputeContext("localpar")
15 mortDS <- RxKdfData("/user/Revoshare/revolution/mortgages", filesystem = hdfsFS)
16
17 # 3. local / stream Hive table
18 rxSetComputeContext("localpar")
19 mortDS <- RxOdbcData(sqlQuery = "select * from mortgages_hive", connectionString = "DSN=Hive", rowsPerRead = 1.0e6)
20
21 # 4. hadoop / HDFS
22 rxSetComputeContext(hadoopCluster)
```

```
1 library(RevoScaleR)
2
3 sqlShareDir <- paste("C:/AllShare/",
4 Sys.getenv("USERNAME"),
5 sep = "\\")
6
7 sqlWait <- TRUE
8 sqlConsoleOutput <- FALSE
9
10 dbConnection <- "Driver={SQL
11 Server};Server=
12 (local);Database=NYCTaxiData;Trust
13 ed_Connections=yes"
14
15
16
17 cc <- RxInSqlServer(
18 connectionstring = dbConnection,
19 shareDir = sqlShareDir,
20 wait = sqlWait,
21 consoleOutput =
22 sqlConsoleOutput)
23
24
25
26
27
28
29
30
31
32
33
34
35
36
37
38
39
40
41
42
43
44
45
46
47
48
49
50
51
52
53
54
55
56
57
58
59
60
61
62
63
64
65
66
67
68
69
70
71
72
73
74
75
76
77
78
79
80
81
82
83
84
85
86
87
88
89
90
91
92
93
94
95
96
97
98
99
100
101
102
103
104
105
106
107
108
109
110
111
112
113
114
115
116
117
118
119
120
121
122
123
124
125
126
127
128
129
130
131
132
133
134
135
136
137
138
139
140
141
142
143
144
145
146
147
148
149
150
151
152
153
154
155
156
157
158
159
160
161
162
163
164
165
166
167
168
169
170
171
172
173
174
175
176
177
178
179
180
181
182
183
184
185
186
187
188
189
190
191
192
193
194
195
196
197
198
199
200
201
202
203
204
205
206
207
208
209
210
211
212
213
214
215
216
217
218
219
220
221
222
223
224
225
226
227
228
229
230
231
232
233
234
235
236
237
238
239
240
241
242
243
244
245
246
247
248
249
250
251
252
253
254
255
256
257
258
259
260
261
262
263
264
265
266
267
268
269
270
271
272
273
274
275
276
277
278
279
280
281
282
283
284
285
286
287
288
289
290
291
292
293
294
295
296
297
298
299
300
301
302
303
304
305
306
307
308
309
310
311
312
313
314
315
316
317
318
319
320
321
322
323
324
325
326
327
328
329
330
331
332
333
334
335
336
337
338
339
340
341
342
343
344
345
346
347
348
349
350
351
352
353
354
355
356
357
358
359
360
361
362
363
364
365
366
367
368
369
370
371
372
373
374
375
376
377
378
379
380
381
382
383
384
385
386
387
388
389
390
391
392
393
394
395
396
397
398
399
400
401
402
403
404
405
406
407
408
409
410
411
412
413
414
415
416
417
418
419
420
421
422
423
424
425
426
427
428
429
430
431
432
433
434
435
436
437
438
439
440
441
442
443
444
445
446
447
448
449
450
451
452
453
454
455
456
457
458
459
460
461
462
463
464
465
466
467
468
469
470
471
472
473
474
475
476
477
478
479
480
481
482
483
484
485
486
487
488
489
490
491
492
493
494
495
496
497
498
499
500
501
502
503
504
505
506
507
508
509
510
511
512
513
514
515
516
517
518
519
520
521
522
523
524
525
526
527
528
529
530
531
532
533
534
535
536
537
538
539
540
541
542
543
544
545
546
547
548
549
550
551
552
553
554
555
556
557
558
559
560
561
562
563
564
565
566
567
568
569
570
571
572
573
574
575
576
577
578
579
580
581
582
583
584
585
586
587
588
589
590
591
592
593
594
595
596
597
598
599
600
601
602
603
604
605
606
607
608
609
610
611
612
613
614
615
616
617
618
619
620
621
622
623
624
625
626
627
628
629
630
631
632
633
634
635
636
637
638
639
640
641
642
643
644
645
646
647
648
649
650
651
652
653
654
655
656
657
658
659
660
661
662
663
664
665
666
667
668
669
670
671
672
673
674
675
676
677
678
679
680
681
682
683
684
685
686
687
688
689
690
691
692
693
694
695
696
697
698
699
700
701
702
703
704
705
706
707
708
709
710
711
712
713
714
715
716
717
718
719
720
721
722
723
724
725
726
727
728
729
730
731
732
733
734
735
736
737
738
739
740
741
742
743
744
745
746
747
748
749
750
751
752
753
754
755
756
757
758
759
760
761
762
763
764
765
766
767
768
769
770
771
772
773
774
775
776
777
778
779
780
781
782
783
784
785
786
787
788
789
790
791
792
793
794
795
796
797
798
799
800
801
802
803
804
805
806
807
808
809
810
811
812
813
814
815
816
817
818
819
820
821
822
823
824
825
826
827
828
829
830
831
832
833
834
835
836
837
838
839
840
841
842
843
844
845
846
847
848
849
850
851
852
853
854
855
856
857
858
859
860
861
862
863
864
865
866
867
868
869
870
871
872
873
874
875
876
877
878
879
880
881
882
883
884
885
886
887
888
889
890
891
892
893
894
895
896
897
898
899
900
901
902
903
904
905
906
907
908
909
910
911
912
913
914
915
916
917
918
919
920
921
922
923
924
925
926
927
928
929
930
931
932
933
934
935
936
937
938
939
940
941
942
943
944
945
946
947
948
949
950
951
952
953
954
955
956
957
958
959
960
961
962
963
964
965
966
967
968
969
970
971
972
973
974
975
976
977
978
979
980
981
982
983
984
985
986
987
988
989
990
991
992
993
994
995
996
997
998
999
1000
```

passenger_count	TotalFares	AvgFares
0	308	43
3	92304	13
4	482841	13
2	2971879	13
1	14959842	12
5	1215883	12
6	832688	12
8	8	8

- Works with RTVS or RStudio
- Speeds up computation locally thanks to MKL
- ScaleR package built-in
- Reproducible R Toolkit built-in
- Work with files on your local file system
- Work with ODBC data from SQL
- Comes with DeployR remote package:
 - Remote into SQL Server and do R on the Server
 - Switch between server and local environments
 - Send R jobs to a server (asynchronously)

Power BI for Data Scientists

Source: R-Scripts

Use R as your source for a Data set in Power BI.

Perform your data preparation in R

Use R models to predict your out-comes and visualize using Power BI

Use R to access ML API in Power BI

R - Plots

Use R-plots in Power BI to visualize your Findings in R.

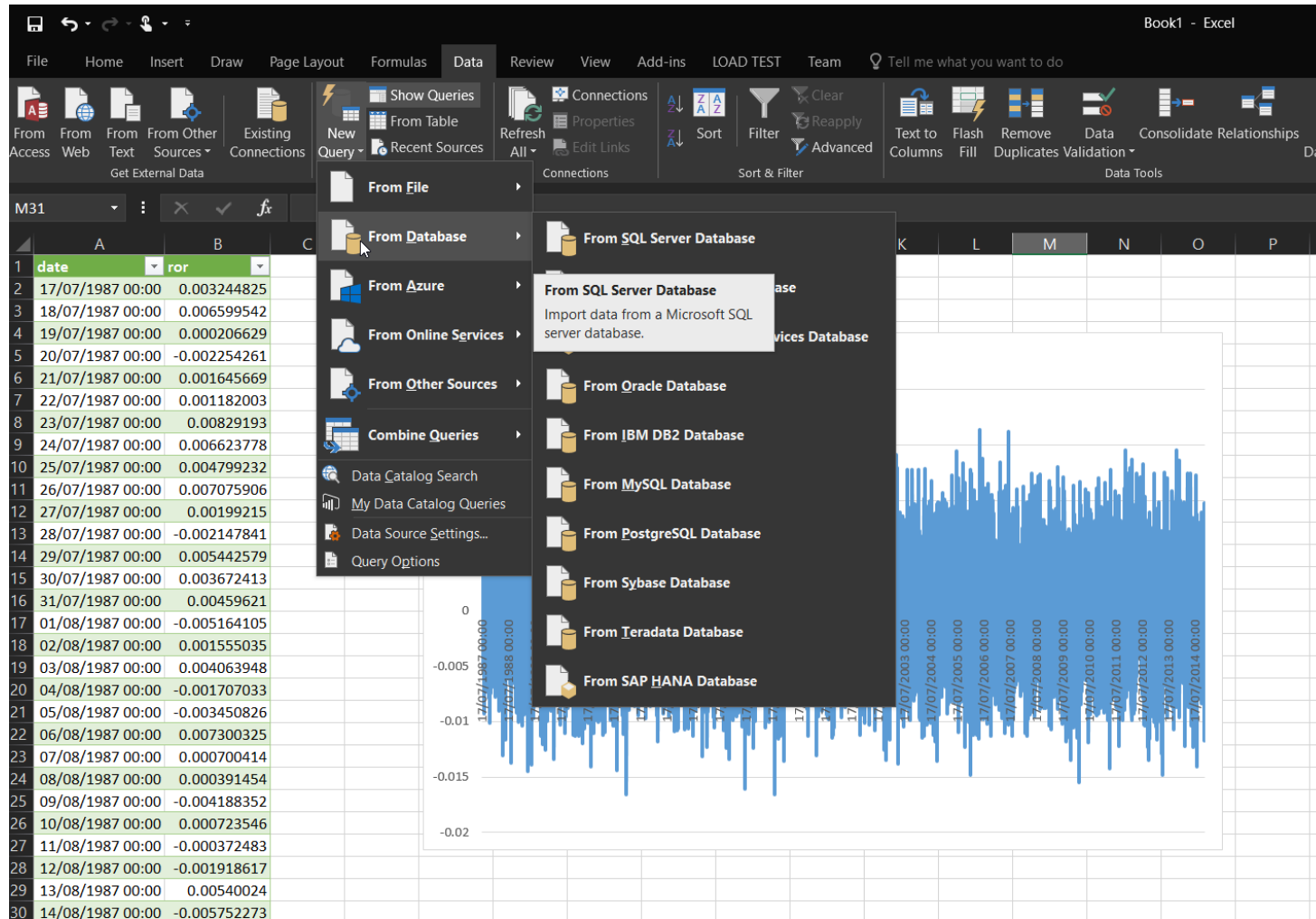
Make your R-Plot slice-able and run Advanced analytics over different Datasets easily

R Driven Visuals

Use R Visuals, without knowing R.

Create your own R-plots for your specific use case

Productivity Tools: Excel Integration



- Allow Excel users to consume your R-based Stored Procedures
- Excel users can consume DeployR Web Services
- No R programming knowledge required for the end user
- Analytics self-service across the organisation

Productivity Tools: SQL Server Reporting

SQL Server Reporting Services

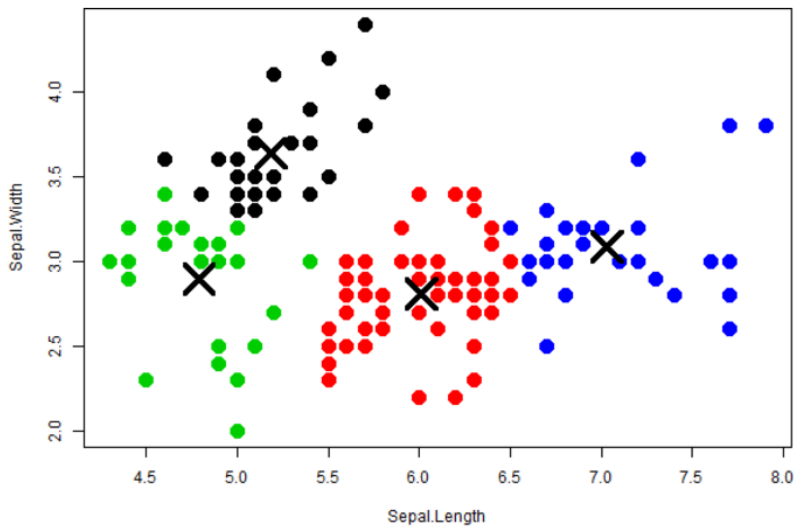
★ Favorites Browse

Home > Simple Example

X Variable Y Variable Cluster Count

1 of 1 100% Find | Next

Iris K Means Example



- On-Premise
- Paginated reports consuming R-based SQL Stored Procedures
- Custom branding
- Drag-and-drop report authoring via Visual Studio or Report Builder
- Real-time prediction
- Built-in Alerting



Data Science track in the Microsoft Professional Program

A MOOC together with Open Edx

Free e Book Data Science with SQL Server 2016

<https://www.r-bloggers.com/free-e-book-data-science-with-sql-server-2016/>

Thank You

The most open platform, ready for production.

LAB 8 – SQL Server + R Services

In this lab you will model the data coming from the New York City taxi trip and fare with SQL Server and MRS.

The data we'll use is a representative sampling of the *2013 New York City taxi trip and fare* dataset, which contains records of more than 173 million individual trips in 2013, including the fares and tip amounts paid for each trip.

To make the data easier and faster to work with for this example, we'll sample it to get just one percent of the data.

You will start creating the SQL context, running a query and ingesting the data.

The modeling task at hand is to predict whether a taxi trip was tipped (a binary, 1 or 0 outcome) based on *features* such as distance of the trip, the duration of the trip, number of passengers in the taxi for that trip, and other factors. Features are columns of data that have a potential relationship to another column, frequently referred to as the *Label or Target*—the answer that we are looking for.

The dataset we have contains past information about the trips, the passengers, and other data (features), and it includes the tip (the label).

LAB 5 – Operationalizing R with AML

In this lab we will explore the functionality of the AzureML R package.

You will learn how to use this package to upload and download datasets to and from AzureML, to interrogate experiments, to publish R functions as AzureML web services, and to run R data through existing web services and retrieve the output.

The AzureML package provides an interface to publish web services on Microsoft Azure Machine Learning (Azure ML) from your local R environment.

The main functions in the package cover:

- Workspace: connect to and manage AzureML workspaces
- Datasets: upload and download datasets to and from AzureML workspaces
- Publish: define a custom function or train a model and publish it as an Azure Web Service
- Consume: use available web services from R in a variety of convenient formats

This lab focuses on small examples rather than trying to solve one particular use case. Therefore, please work through the examples and exercises.

Lab 7: MRS on HDI Premium (Spark)

1. Import data from CSVs into Spark DataFrames
2. Cleaning and manipulating Spark DataFrames with SparkR
3. Exporting Spark DataFrames to XDF
4. Training models using ScaleR functions in Spark compute context
5. Deploying models trained on 'big' data and operationalizing them in the cloud with Azure ML

Typical advanced analytics lifecycle

Prepare: Assemble, cleanse, profile and transform diverse data relevant to the subject.

Model: Use statistical and machine learning algorithms to build classifiers and make predictions

Operationalize: Apply predictions and visualizations to support business applications

