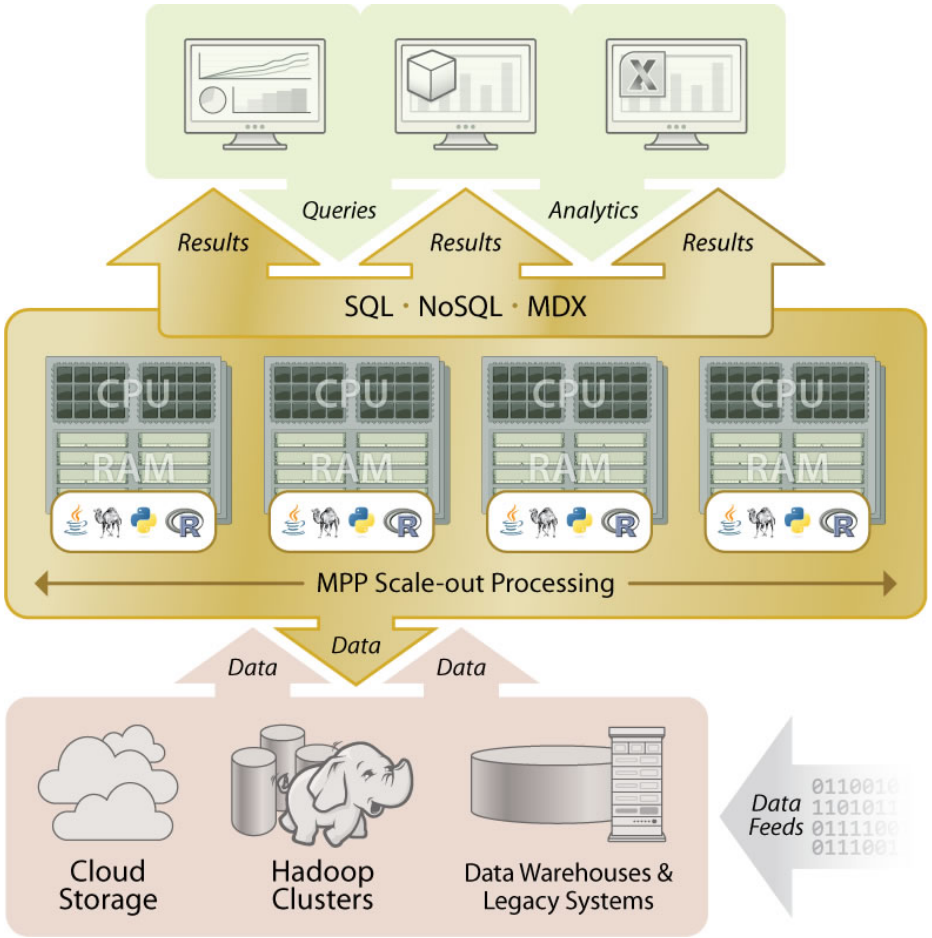
Understanding Kognitio RAM

including managing RAM for external scripts

RAM management

Kognitio Architecture Overview

Application & Client Layer



Kognitio

Analytical Platform Layer

Persistence Layer

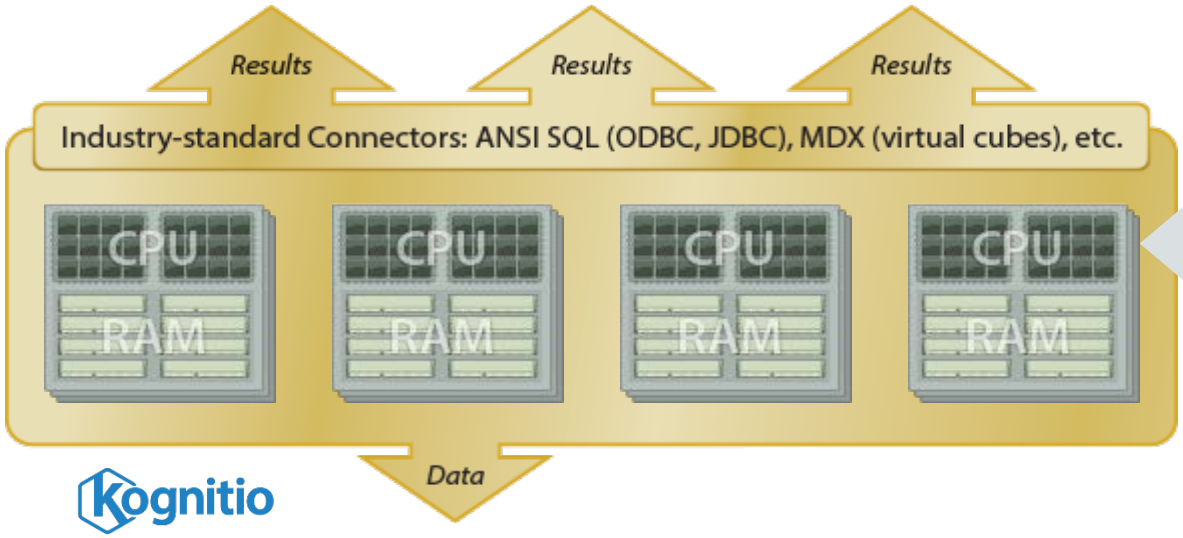
Kognitio performance comes from the full exploitation of ALL CPU cores on a system (full MPP) AND in-memory processing. All data during processing is held in RAM

It is therefore important to appreciate how RAM is utilised in a Kognitio system

Kognitio

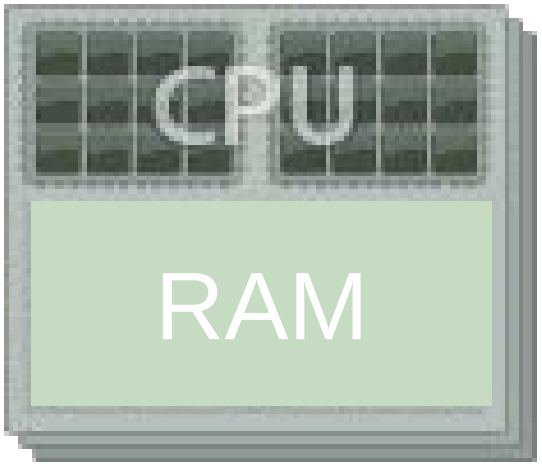
RAM management

Kognitio node interaction



Kognitio is made up of a series of **nodes** each containing a set of CPU cores and RAM.

Kognitio handles all interaction **between** these nodes automatically. This includes passing data, collating result sets etc

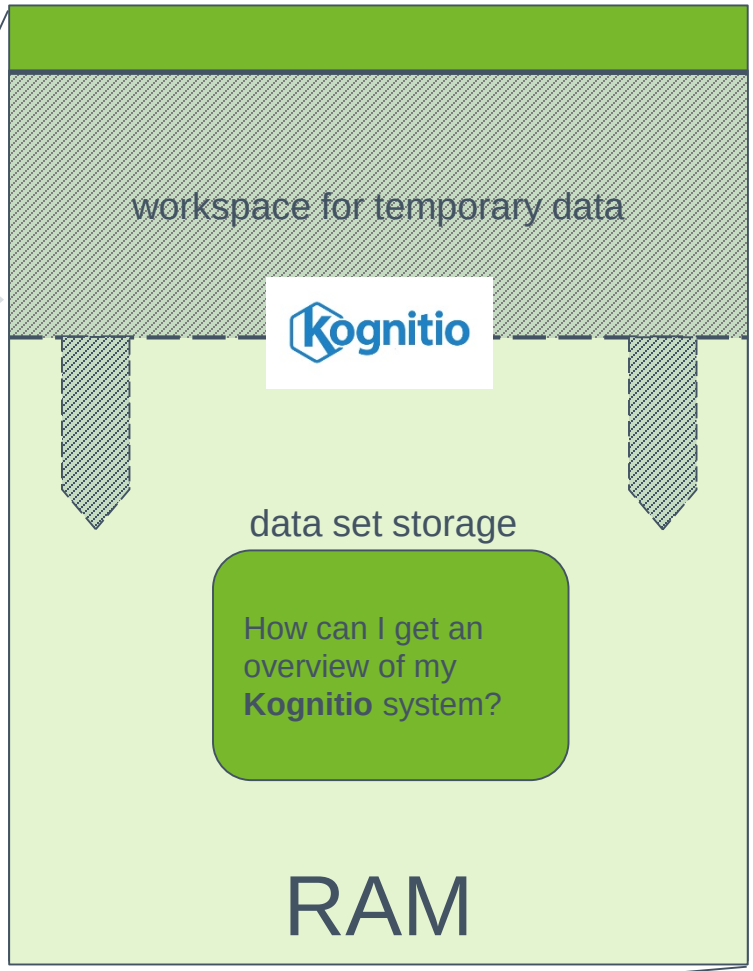


This module concentrates on RAM utilisation on one node but remember all nodes on the system have a similar set up.

Kognitio

RAM management

Kognitio default configuration



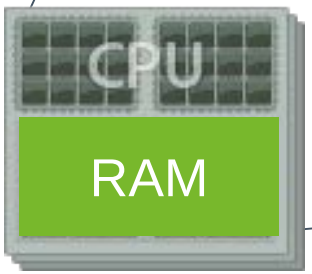
When a Kognitio system is configured 7% of available RAM is set aside for processes **external** to Kognitio (including external script invocations)

Kognitio utilises the remaining 93% of RAM.

Within Kognitio 30% of RAM is always reserved as **workspace** for holding **temporary data** when running SQL queries.

Within Kognitio the remaining 70% is available for “pinning” **data sets** into memory

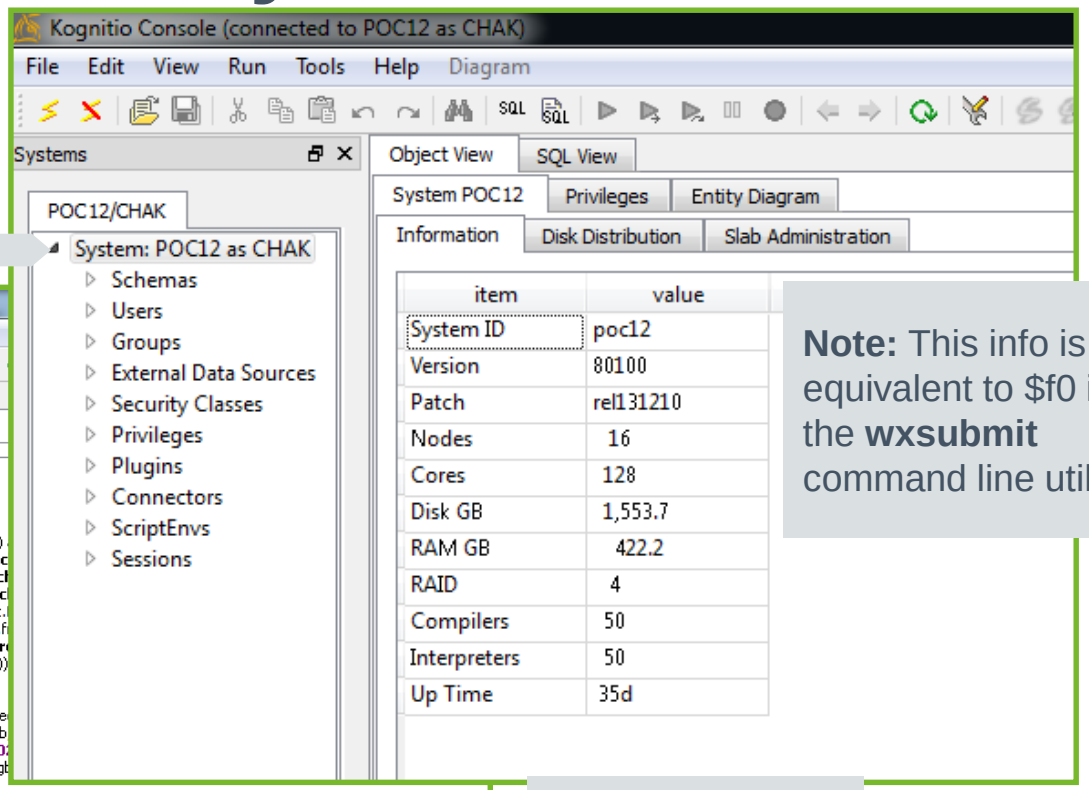
Note when data set storage is not used it is automatically exploited as workspace

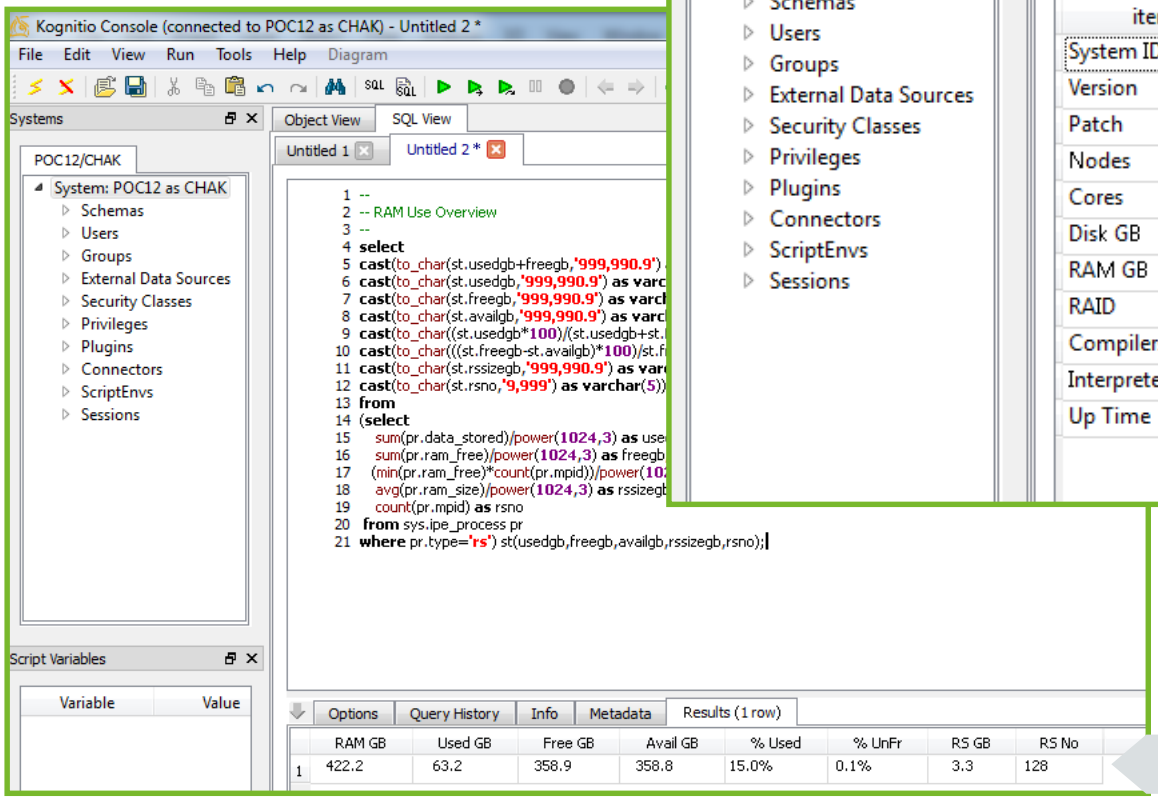


Kognitio

RAM management
Viewing Kognitio system overview

In Kognitio console double click on System in the Systems tree pane and select the Information tab.



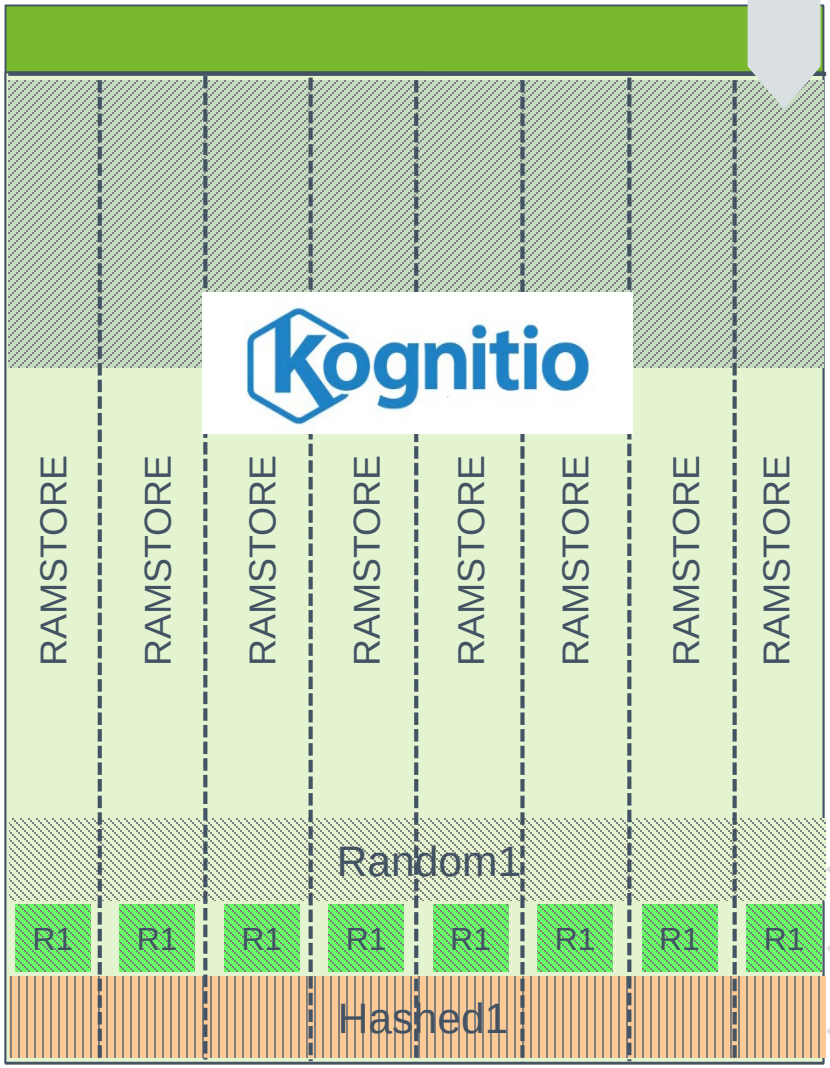


Note: This info is equivalent to $f0 in the **wxsubmit** command line utility

The ctrl-F2 command in console also contains useful information about system RAM

RAM management

Kognitio RAM stores for MPP



During system configuration the Kognitio RAM on a node is automatically divided into **ramstores.** Typically 1 ramstore is created for **every 4GB of RAM**

Note: **Ramstore** info is also displayed in RAM overview

Any table or view images created are pinned into Kognitio RAM. **Default** distribution is **random**

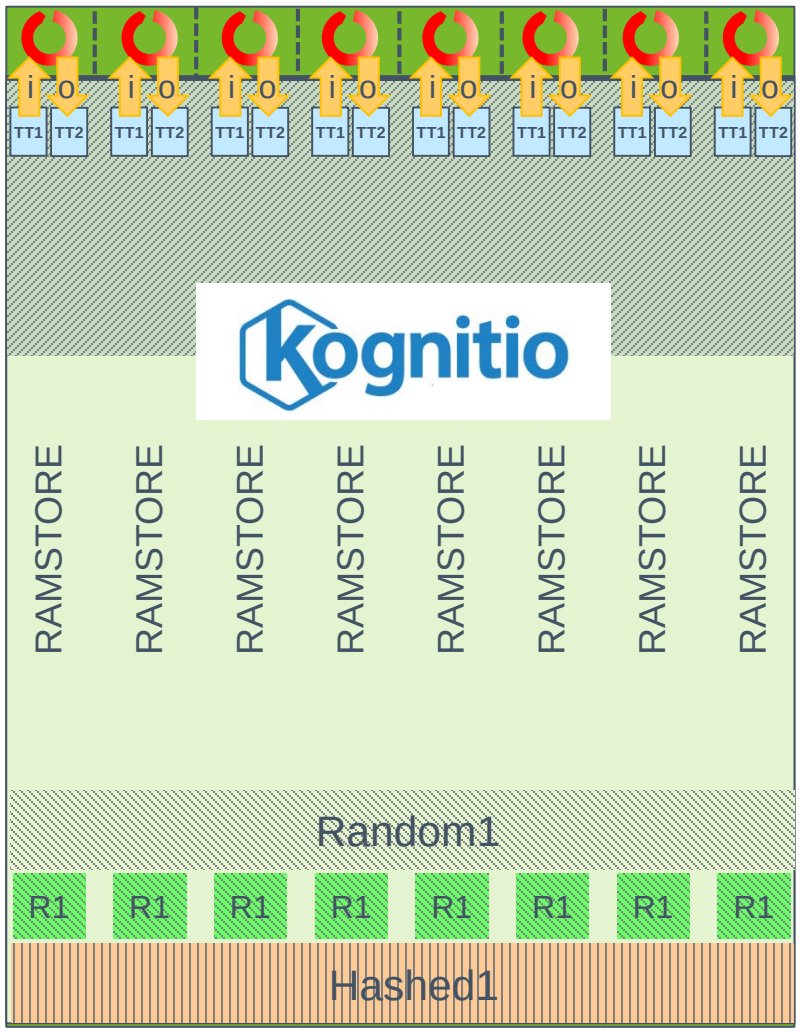
For small **lookup** data sets consider **replicating** to every **ramstore** to improve join performance

Hashed images ensure rows with same hash values are located in the same **ramstore;** ideal for frequent large table joins

Kognitio

RAM management

Extending MPP to external scripts



Default parallelism means that one script process is invoked per **ramstore**.

All processing and interim results **within** the script must be held in the RAM **external** to Kognitio

When an external script query is submitted input data from a temporary table in the **parent ramstore** is streamed into the script invocation. Output is streamed back to a temporary table.

The default RAM limit for a script is 100 MB. This ensures 2 external script queries can run concurrently with default parallelism.

What happens if external script memory requirement is too large?

Kognitio

RAM management

External script RAM requirement is too high

An external script will fail if the RAM requirement is too high. Users will see different errors depending on where in the script the process fails. Here the entire input data set could not be read in by R

```
86 --
87 -- Run the same query over all data by removing predicate
88 --
89 select * from
90 (external script Kog_R_AvgOverAllRows from
91    (select prodno, price/100.00 from demo_ret.v_ret_sale
92 --        where storeno = 1
93       )
94 ) ext;
```

Error | Results (0 rows) | Metadata | Info

Error: Go To Error

HY000[Kognitio][WX2 Driver][POC12] RS0023: Error writing to external script pipe

Info:
Data could not be properly sent to the external script. It may have exited without consuming all data.

Action:
Check the server logs and debug the script.

Checking the logs will indicate memory allocation errors such as when R cannot allocate space for a vector.

```
1
2 external script DEMODATA.DEBUG_LOG
3 order by 1 desc;
```

Results (100 rows) | Metadata | Info

	LOG_OUTPUT
1	T_2014-01-27_10:35:12_GMT: AM id 0x2f00000f: abort request receive…is AM. session=645119, tno=-1, session aborted…
2	T_2014-01-27_10:35:12_GMT: AM #2f00000f aborting now
3	T_2014-01-27_10:34:52_GMT: RS 0 S 645000 R 33700B8 LO:Script std…xecution halted
4	T_2014-01-27_10:34:52_GMT: RS 0 S 645000 R 33700B8 LO:Script stderr: Error: cannot allocate vector of size 250.0 Mb
5	T_2014-01-27_10:34:52_GMT: RS 0 S 645000 R 33700B8 LO:Nonzero script exit code 0x40000001
6	T_2014-01-27_10:34:52_GMT: RS 0 S 645000 R 33700B8 LO:Error writing to pipe, ecode 32.

To access the log files you need either wxadmin access to any Kognitio node or have an external script set up to view the logs. Learn how here.

Kognitio

Further content to follow soon

(and accompanying examples)

- RAM management at script level – for end users
  - Threads and requires RAM settings
- RAM management at environment level – for administrators
  - Threads and requires RAM settings
- Making more RAM available to external scripts
  - Kognitio parameters
  - system configuration

Kognitio

connect

www.kognitio.com

NA: +1 855 KOGNITIO
EMEA: +44 1344 300 770

linkedin.com/company/kognitio

twitter.com/kognitio

Facebook.com/kognitio

youtube.com/kognitio

Kognitio