T-SQL & CLR II

More ways to use the CLR in SQL Server

Overview

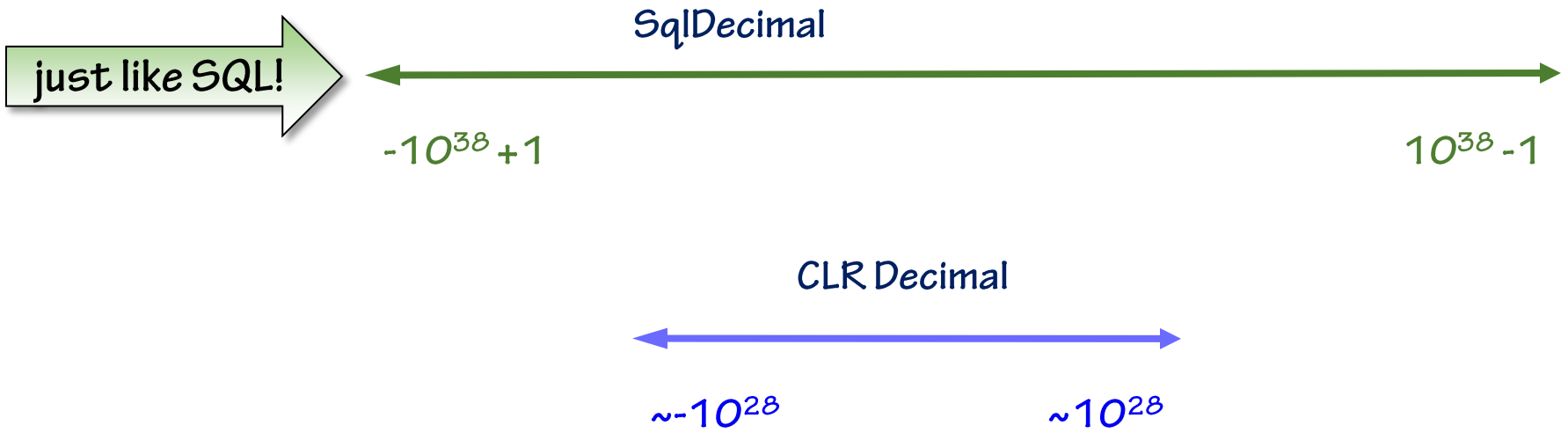
- SqlTypes
- User defined process
- User defined aggregate

SqlTypes

- .NET equivalents
  - "make sense" to SQL Server
- Match SQL Server types
  - representation
- Null semantics

Representation

- SqlDecimal matches representation of Decimal on SQL Server

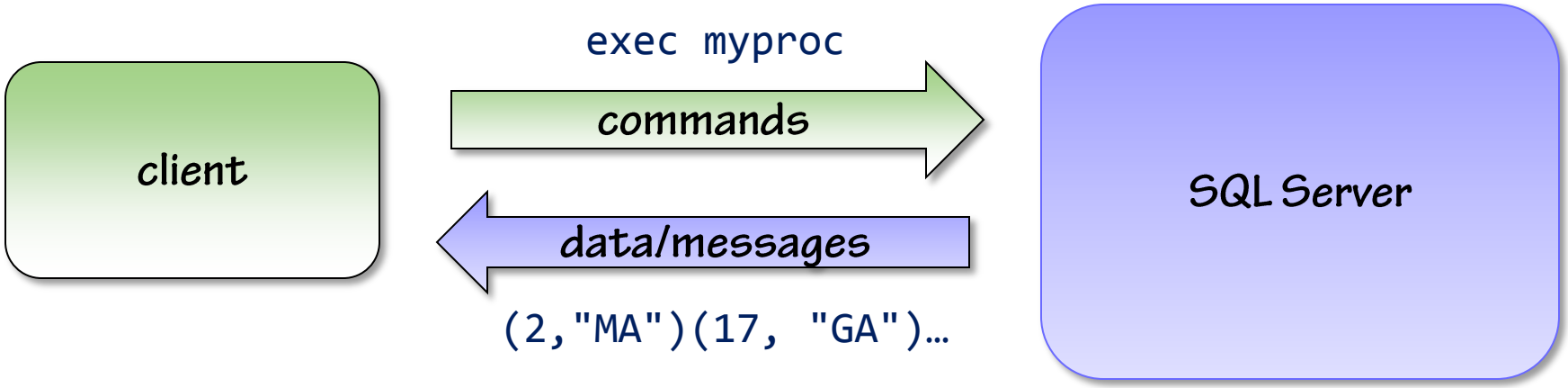


SqlString

- Default 4000 chars (vs)
- UCS-2 vs UTF-8

User defined process

- SqlProcedure
- Return status
- Input and output
- Connected via pipe to client

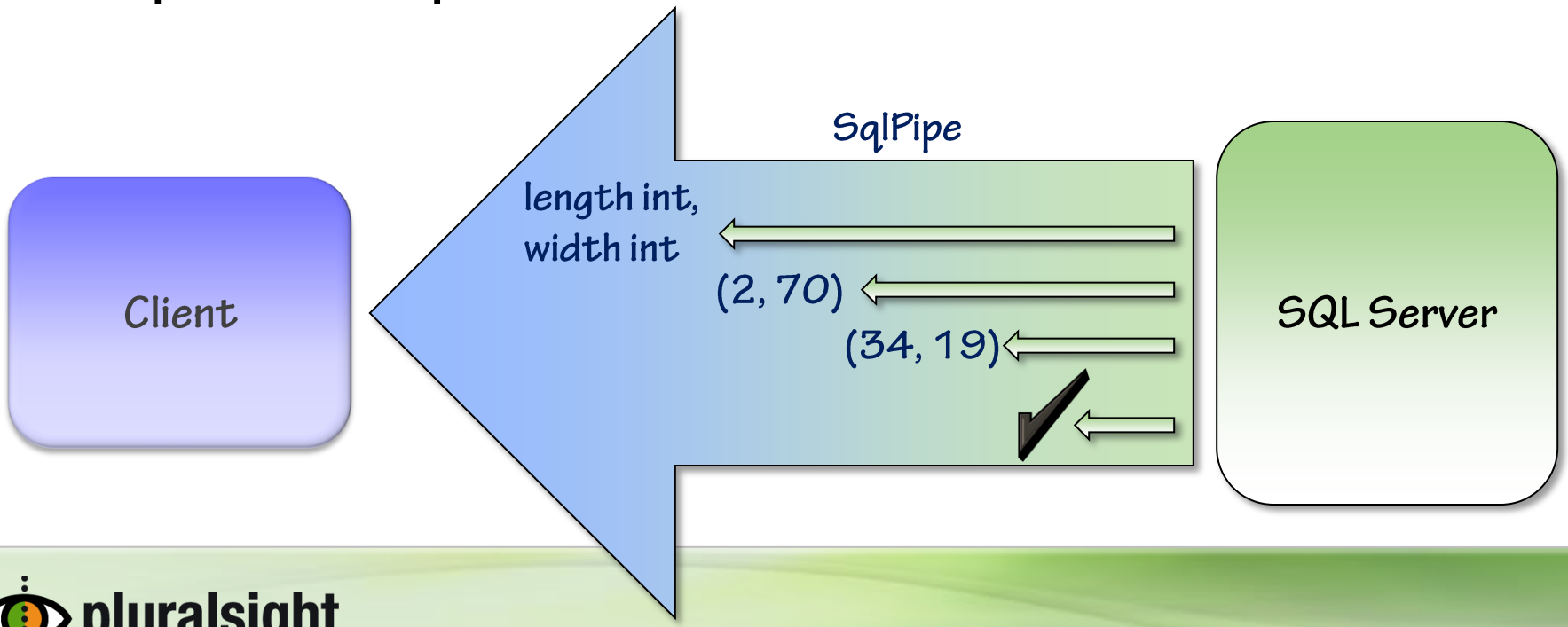


SqlContext

- **Provides access to SqlPipe**
  - server connection to TDS stream to client
- **Messages**
- **Resultsets**
  - constructed
  - executed
- **Errors**

Constructed Resultset

- Metadata, *i.e.* the columns must be defined
- Send metadata to client
- Return rows to client
- Mark end of resultset
- Repeat for multiple resultsets



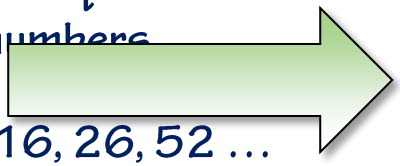
Fibonacci Series

- Needs two numbers two start it
- Subsequent numbers are the sum of the two that precede it

starter numbers: 4, 6

subsequent numbers: 10, 16, 26, 52 ...

```
exec dbo.MakeFib(4, 6, 6)
```

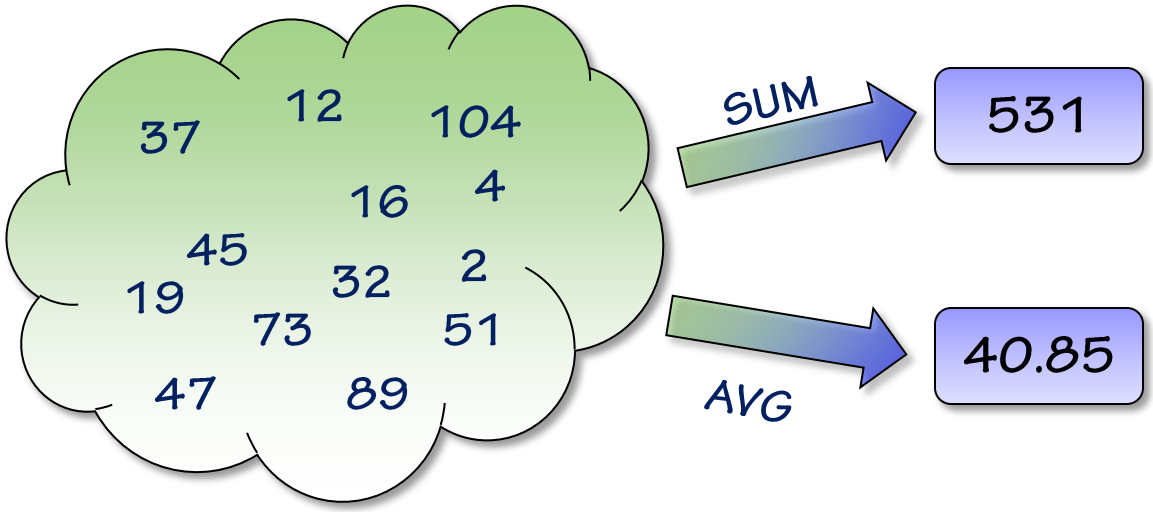


position	value
1	4
2	6
3	10
4	26
5	52
6	78

Aggregate

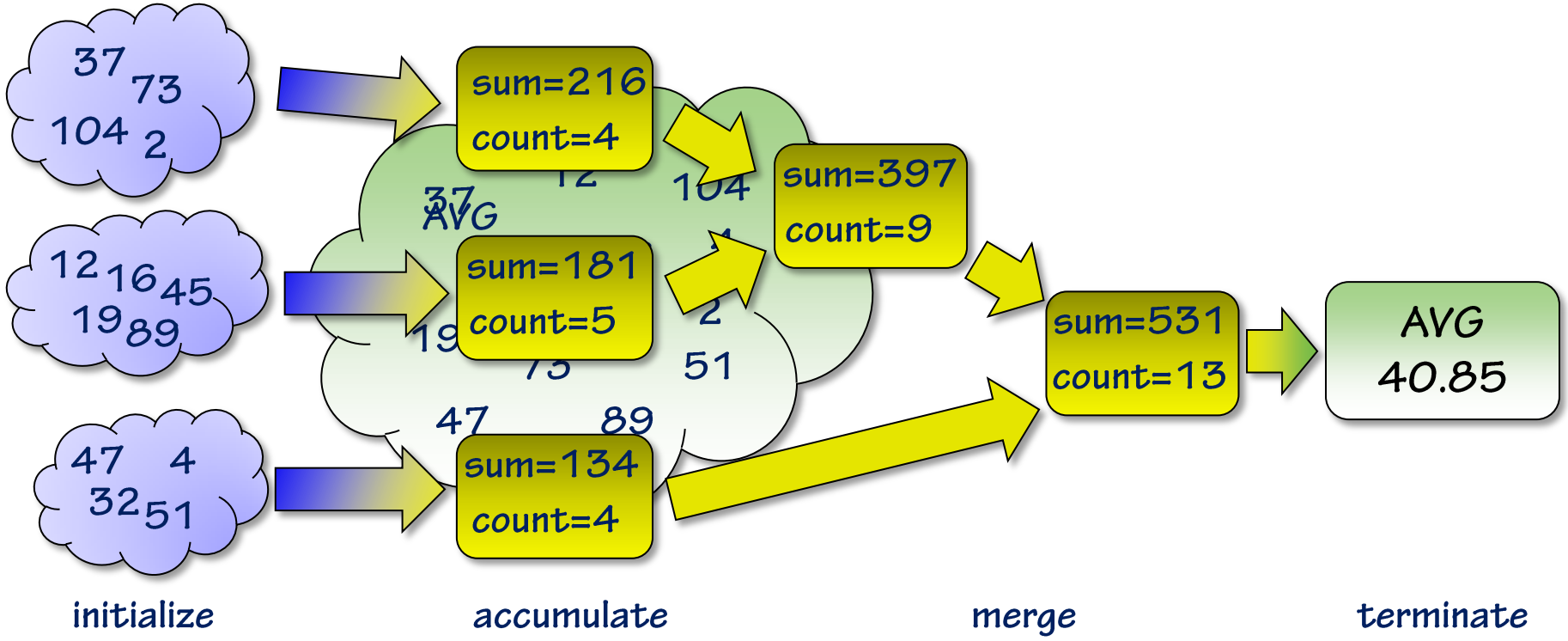
- Produces scalar result from set of data
- Typically better performance than cursor

```
select sum(2*(width+height)) from dbo.Rectangles;
```



SqlUserDefinedAggregate

- Public class
- Accumulated state… limited
- Lock free concurrency



Summary

- **SqlType's have the semantics of SQL Server types**
- **SqlProcedure marks static public member of public class**
  - input and output parameters
  - SqlPipe to send information back to client
  - access database including making changes
- **SqlUserDefinedAggregate marks public class**
  - performance on order of builtin aggregate
  - use instead of implementing cursor

References

- **Extras**
  - http://pluralsight-free.s3.amazonaws.com/dan-sullivan/SQLCourseExtras/CLR2.zip