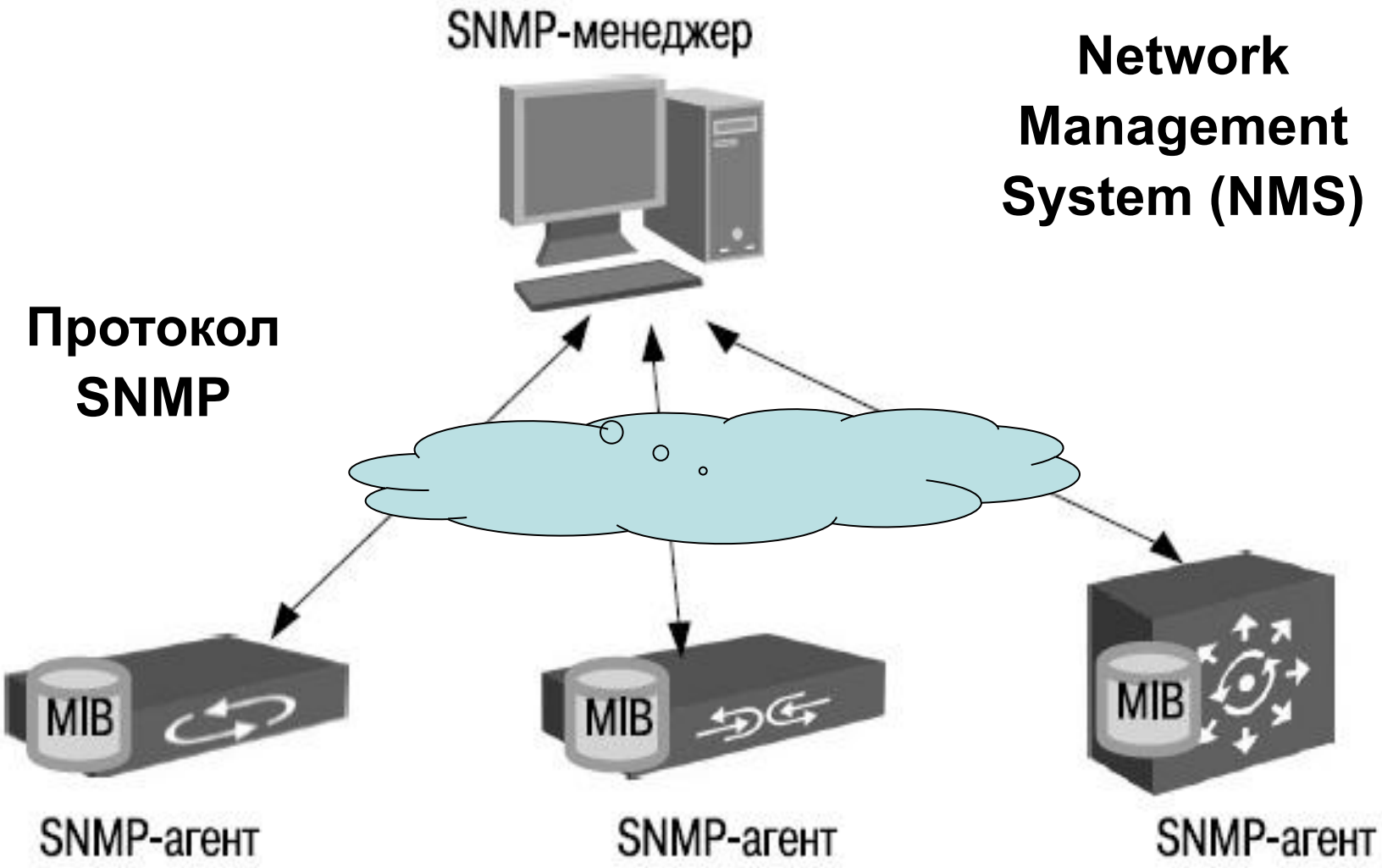
Моделирование протокола SNMP с помощью пакета Telelogic TAU SDL

Архитектура системы управления



Программно-аппаратные средства менеджера размещаются в системе управления сетью (Network Management System – NMS).

На каждом удаленном сетевом устройстве располагается агент, который:

- снабжает менеджера данными о текущем состоянии устройства;
- предоставляет менеджеру возможность ***косвенно*** управлять этим устройством, изменяя его состояние.

Обмен управляющей информацией между менеджером и агентами

Организация MIB

MIB (Management Information Base) – это база управляющей информации

Здесь хранятся значения переменных (Values of Variables), которые представляют интерес для определенных задач управления

К этим переменным относятся параметры (характеристики), которые контролируются в процессе управления и на которые можно оказывать воздействие

Примеры таких характеристик:

- физический адрес сетевого интерфейса;
- число пакетов, переданных через некоторый интерфейс;

Для именования всех переменных в составе MIB предназначен идентификатор объекта **OID** (Object Identifier)

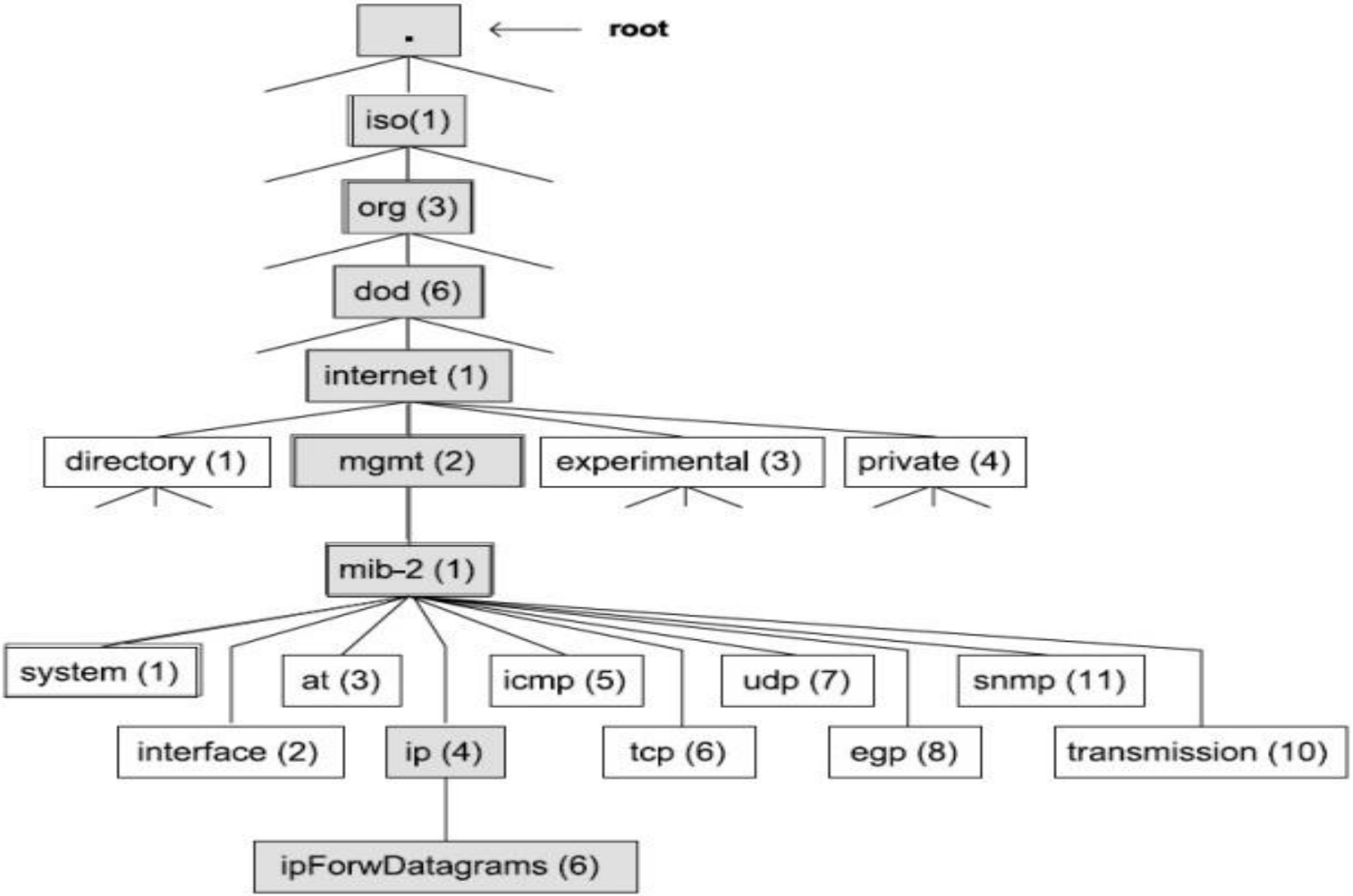
При этом каждая переменная имеет уникальное значение OID

Пространство таких идентификаторов построено по ***иерархической схеме***

В результате обеспечивается жесткая структура идентификатора объекта, а также ускоряется доступ к данным

По этой причине логическую структуру MIB часто изображают в виде ***дерева***

Иерархическое представление объектных идентификаторов



Каждому узлу дерева приписано целое число – порядковый номер в пределах некоторой группы

К этому номеру можно добавить весь путь через предшествующие узлы верхних уровней

Тогда получится значение OID в ***цифровом*** виде

1.3.6.1.2.1.1

Человеку удобнее работать с ***текстовым*** представлением отдельных элементов OID

Например, для получения общей информации о системе необходимо пройти по ветке

В этом случае эквивалентное значение OID для общей информации о системе имеет вид:

iso.org.dod.internet.mgmt.mib-2.system

Типы данных

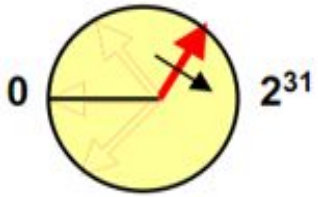
1) Простые типы

```
IpAddress ::=   OCTET STRING (SIZE (4))
```

160.85.128.1

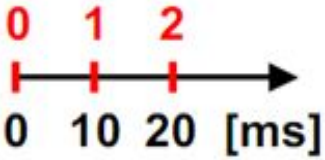
A0 55 80 01

```
Counter   ::=   INTEGER (0..4294967295)
```



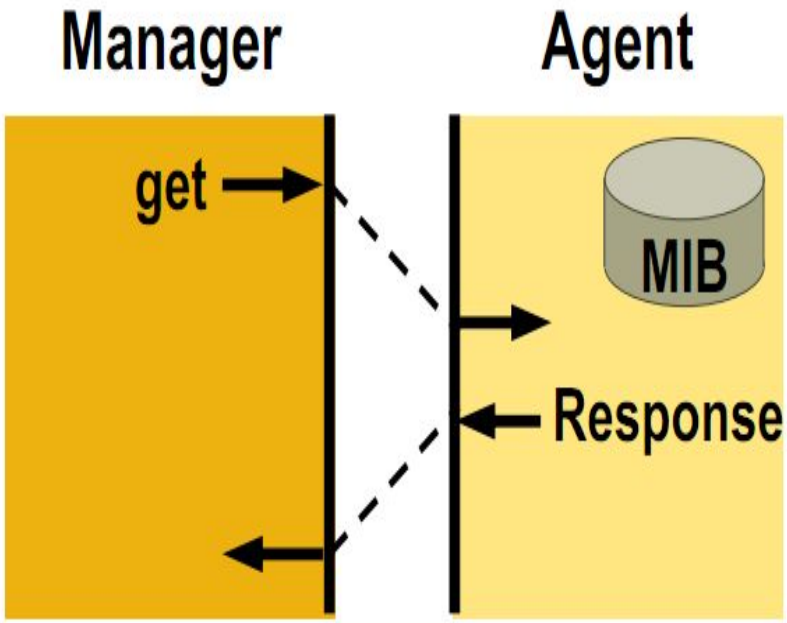

```
Gauge     ::=   INTEGER (0..4294967295)
```

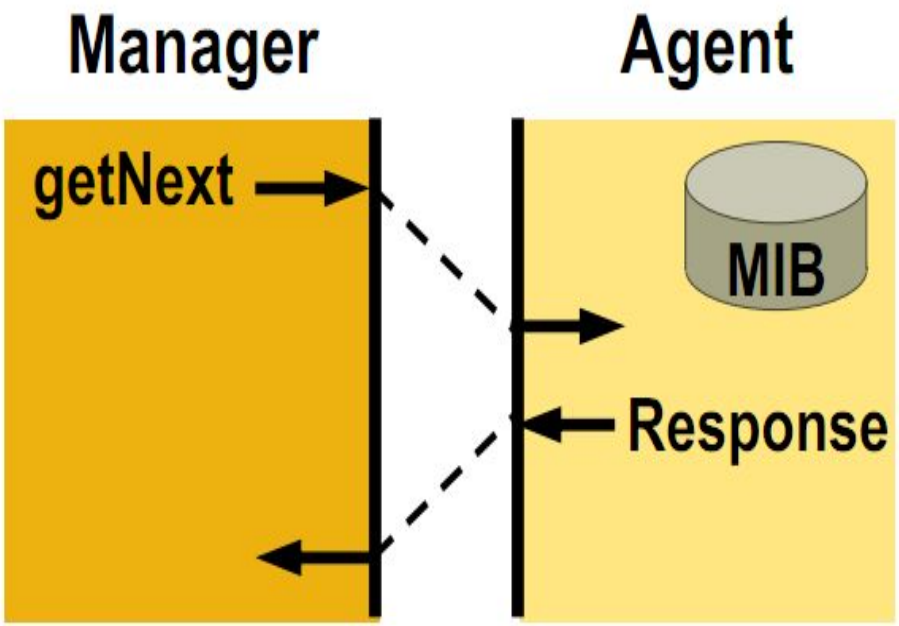
```
TimeTicks ::=   INTEGER (0..4294967295)
```




```
Opaque    ::=   OCTET STRING
```

Операции управления

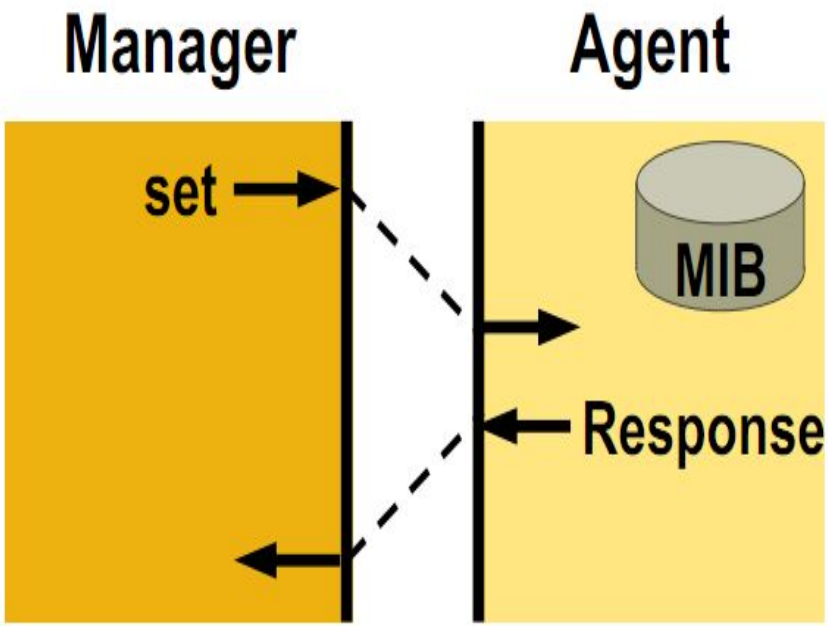


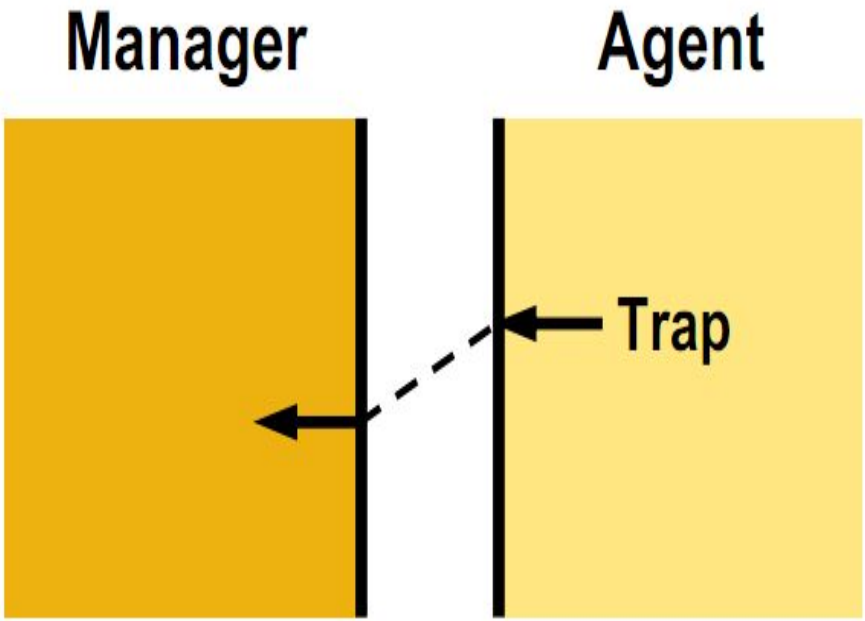


Позволяет получить текущее значение некоторых переменных из MIB

Позволяет получить текущее значение следующей по порядку переменной из MIB

Операции управления (продолжение)





Позволяет установить новые значения для некоторых переменных из MIB

Позволяет агенту самостоятельно сообщать о проблемах, связанных с объектом управления

Общий формат сообщений протокола SNMP

Version	Community	SNMP PDU

Version – целое число (Integer), которое на 1 меньше применяемой версии протокола

Community – символьная строка (Octet string) с названием некоторой группы (сообщества) агентов и менеджеров.

Фактически играет роль ***пароля*** для доступа к соответствующим управляемым ресурсам.

PDU (Protocol Data Unit) – основная часть

Форматы основной части сообщения (PDU)

Сообщения Get, GetNext, Set, Trap

PDU type	Request-id	0	0	Variable-bindings

Сообщение Response

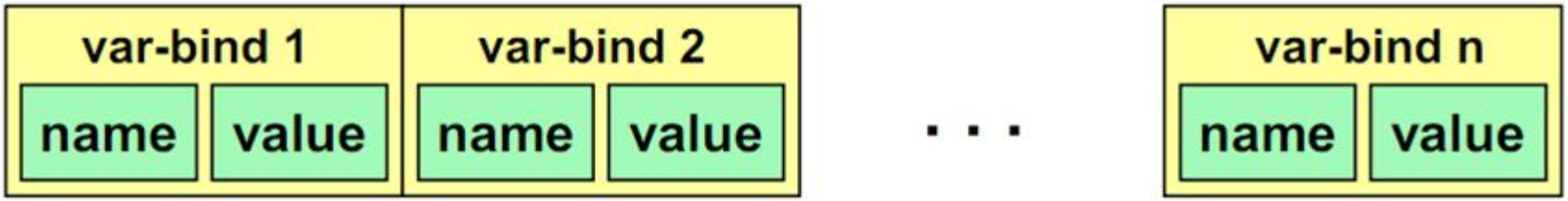
PDU type	Request-id	Error-status	Error-index	Variable-bindings

PDU type – тип сообщения (целое число от 0 до 4)

Request-id – порядковый номер запроса; целое число, позволяющее логически связать конкретный запрос от менеджера и ответы от агента, относящиеся к этому

Variable-bindings – информационное поле с типом данных ***Sequence***.

Содержит связки (пары) «**name** & **value**» для определенных переменных из переменной.



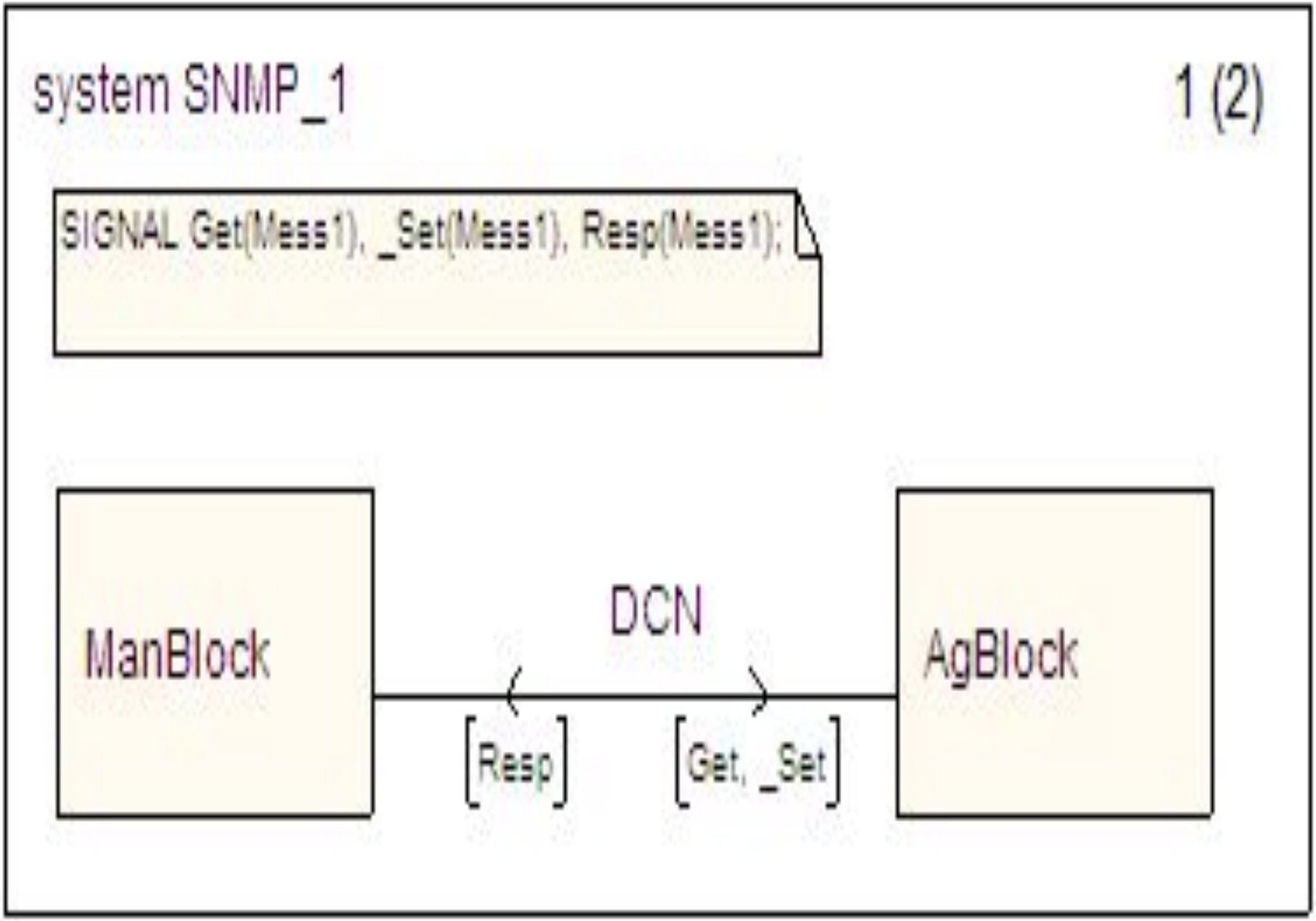
В качестве **name** используется **OID**

Error-status – код ошибки, значения которого соответствуют следующим ситуациям:

- **0** (**noError**) – ошибок нет
- **1** (**tooBig**) – слишком большой объем данных

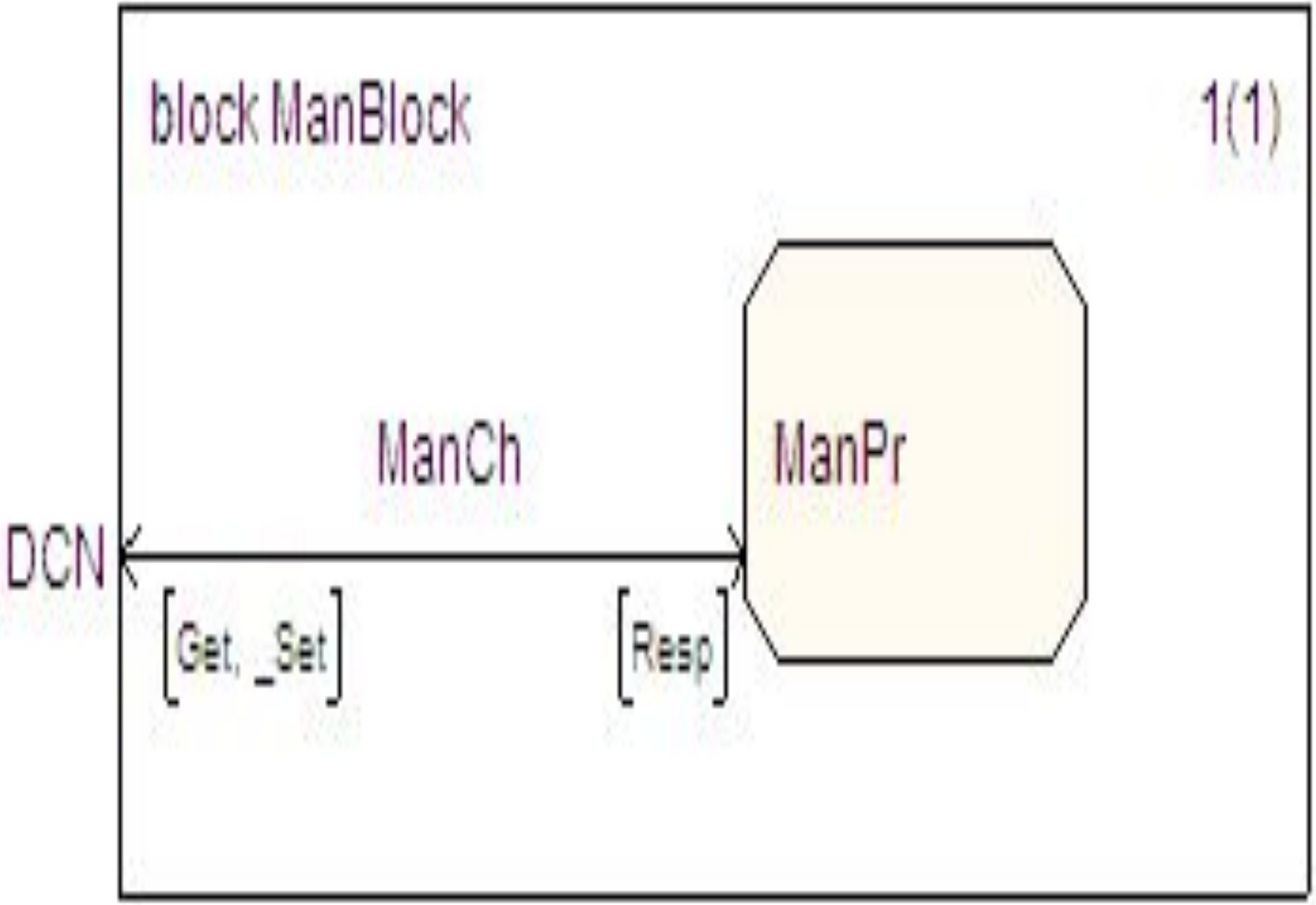
Error-index – при **errStatus**≠0 указывает номер переменной в списке **varBinds**, к которой относится ошибка

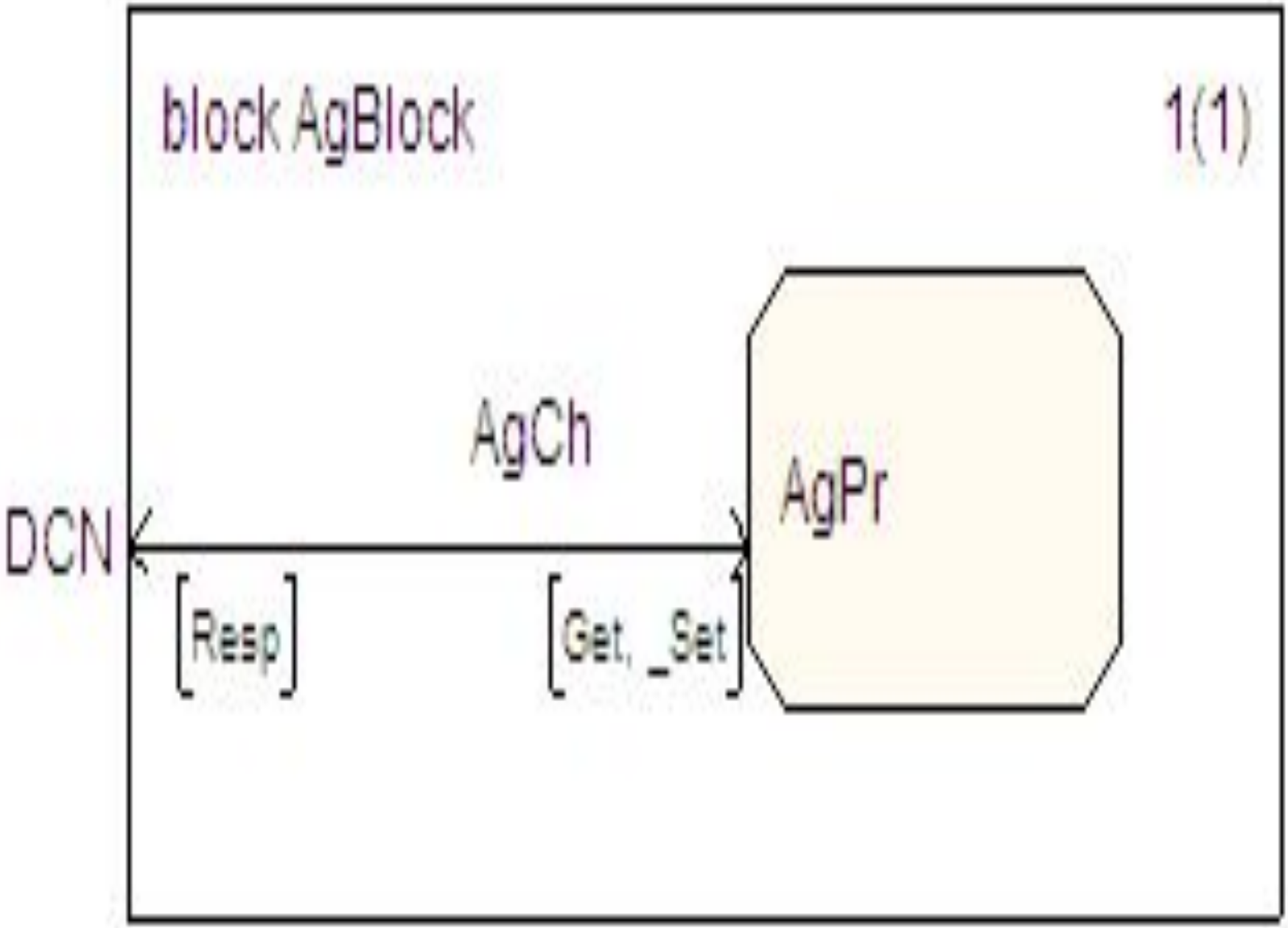
Модель протокола SNMP на языке SDL


system SNMP_1
1 (2)
SIGNAL Get(Mess1), _Set(Mess1), Resp(Mess1);
ManBlock
DCN
AgBlock
[Resp]
[Get, _Set]

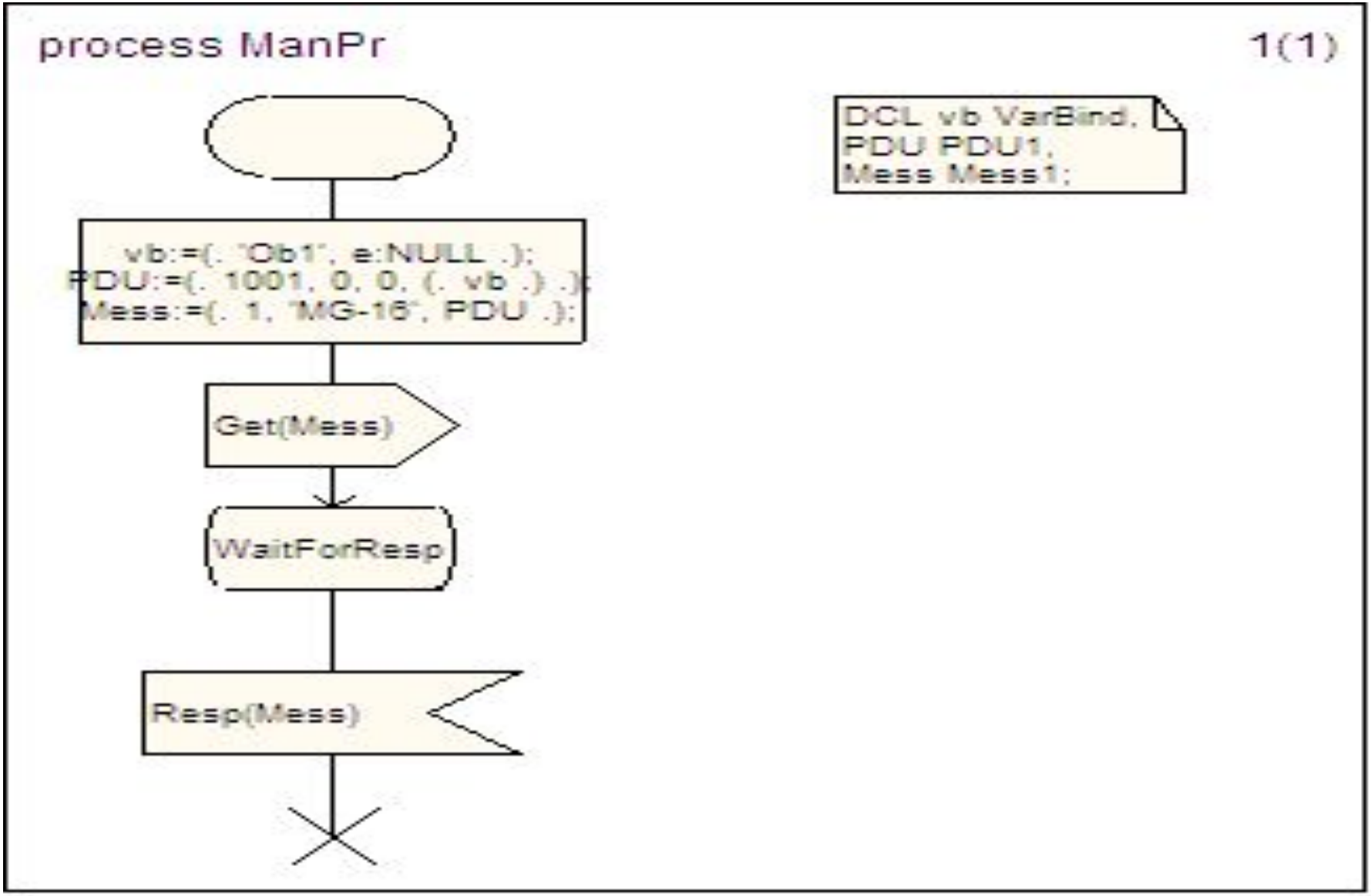
system SNMP_1


```
newtype val_type choice
  a Integer;
  b Charstring;
  e NULL
endnewtype;

newtype VarBind struct
  name Charstring;
  value val_type
endnewtype;

newtype VBList
  String(VarBind, emptystring)
endnewtype;

newtype PDU1 struct
  req_id Integer;
  err_status Integer;
  err_index Integer;
  var_binds VBList
endnewtype;

newtype Mess1 struct
  version Integer;
  community Charstring;
  data PDU1
endnewtype;
```

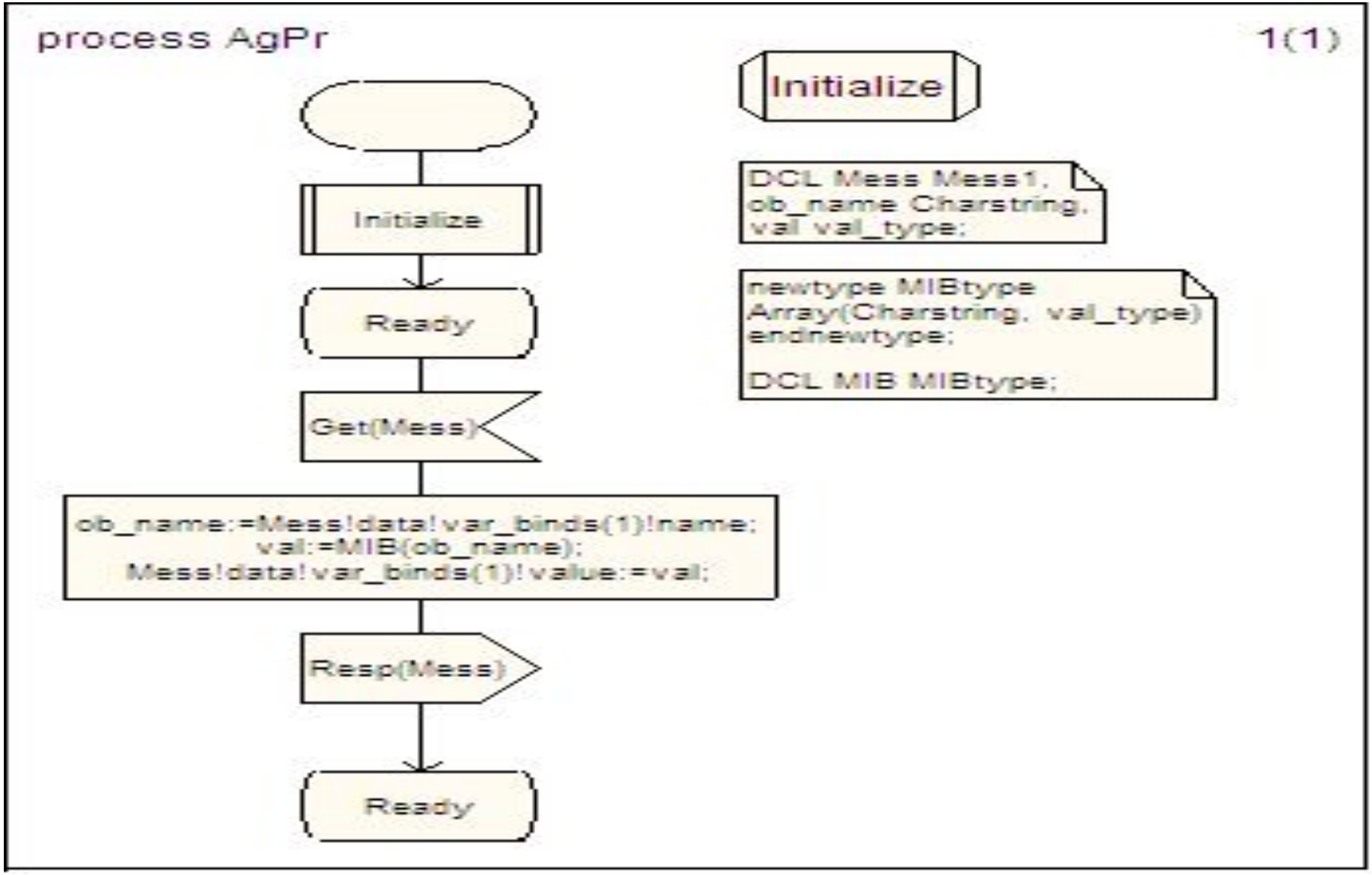

block ManBlock
1(1)
ManCh
ManPr
DCN
[Get, _Set]
[Resp]


block AgBlock
1(1)
AgCh
AgPr
DCN
[Resp]
[Get, _Set]

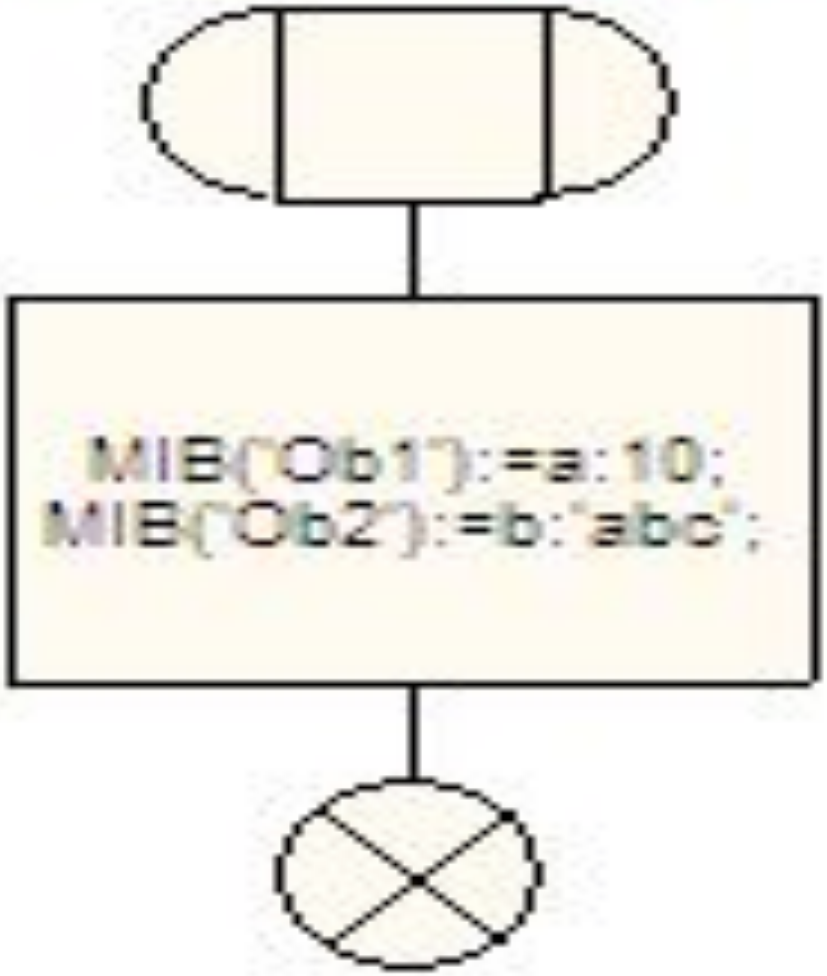
Запрос Get с одной переменной



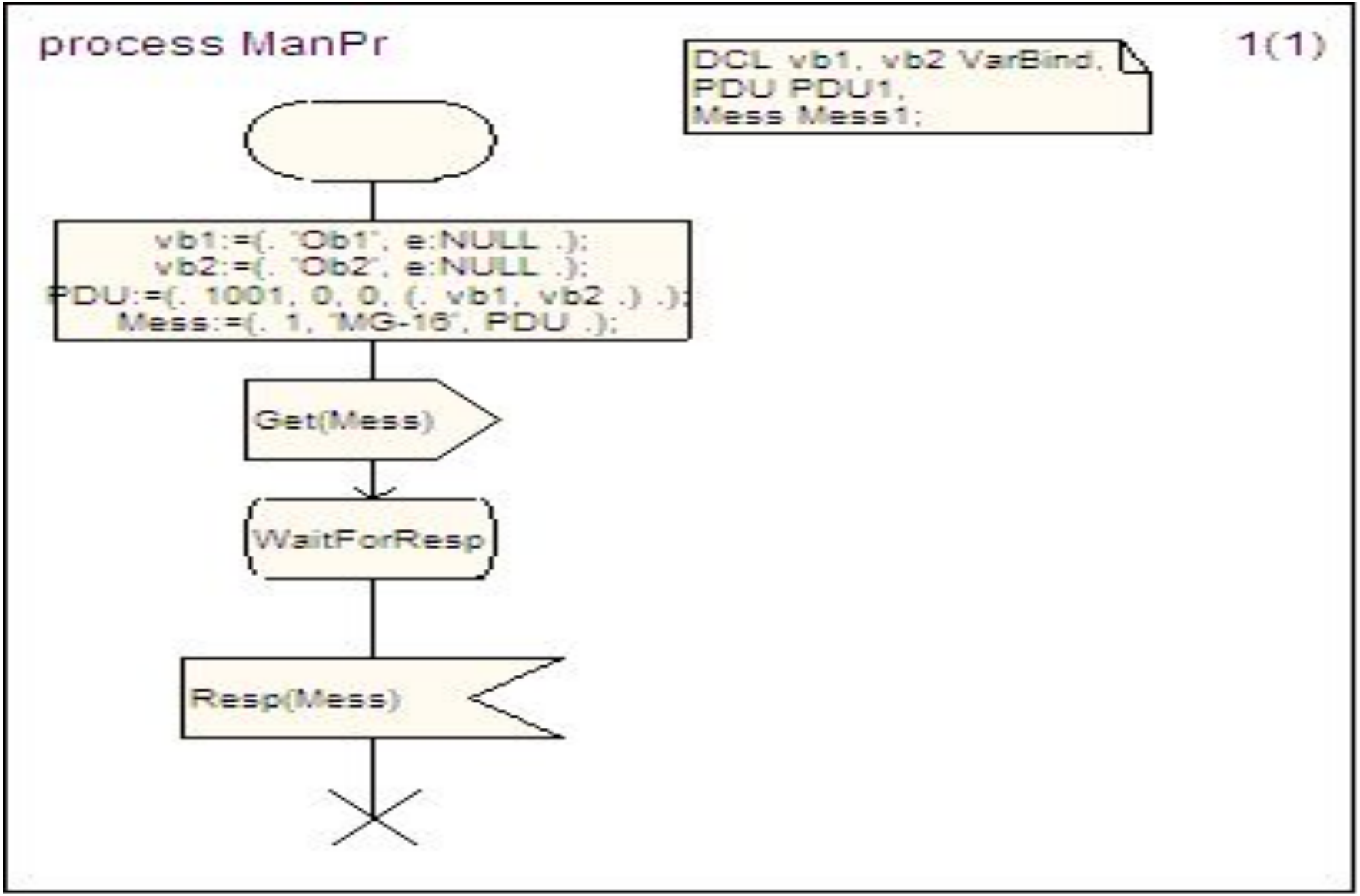
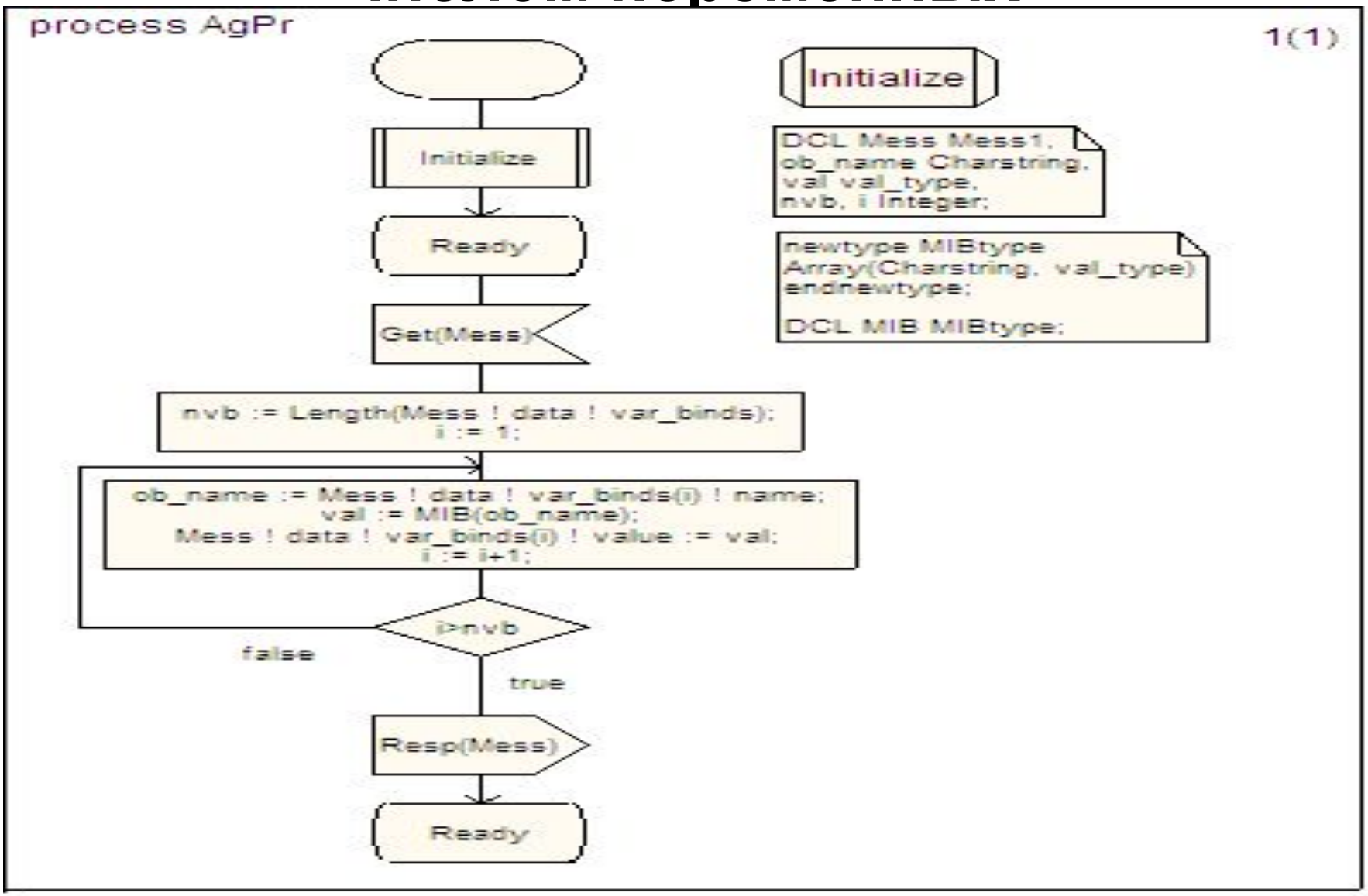
Ответ Response с одной переменной



procedure Initialize 1(1)



Запрос Get с двумя переменными



Ответ Response с произвольным числом переменных