

## 1) Набор новых функций

### - GetMoniker

Идентификация текущего объекта для передачи в скрипты и внешние программы. На входе номер объекта, на выходе - строка идентификации объекта (моникер).

```
<строка моникера> = GetMoniker(<число objid>);
```

### - GetIDByMoniker

Идентификация возвращаемых объектов обратно из скриптов и внешних программ. На входе - строка идентификации объекта (моникер), на выходе - номер объекта для использования в SetObjID.

```
<число objid> = GetIDByMoniker(<строка моникера>);
```

### - SetSubscription или Subscription

Установка подписки. На входе - номера объектов, первый - тот, который будет оповещать о своих событиях, второй - тот, который будет получать оповещения.

```
SetSubscription <число objid1> <число objid2>;
```

### - UnSubscription

Сброс подписки. На входе - номера объектов, первый - тот, который оповещал о своих событиях, второй - тот, который получал оповещения.

```
UnSubscription <число objid1> <число objid2>;
```

### - ClearSubscriptions

Сброс подписки. На входе - номер объекта, кто оповещал о своих событиях.

```
ClearSubscriptions <число objid1>;
```

- GetSubscriptions

Получение адресатов подписки. На входе - номер объекта, который оповещает о своих событиях. Результат записывается в текущий активный буфер без сброса его содержимого в виде строк, содержащих номера объектов.

```
GetSubscriptions <число objid1>;
```

2) Для каждого объекта, который будет получать оповещения, нужно определить свойство (численное с номером или строковое с именем алгоритма), которое определит алгоритм, вызываемый системой оповещения в конце транзакции.

Все изменения объектов, происшедшие в процессе выполнения алгоритма, не будут вызывать оповещения.

- OnChangeProps (согласовать!!!) - номер или строка имени алгоритма, вызываемого при изменении свойств объекта. На входе в алгоритм текущим является информируемый объект, а параметром алгоритма передётся номер объекта, информирующий о своём изменении и вызвавший этот алгоритм.

3) Система оповещения создаст файл алгоритма NOTIFY.ALG во временной папке ADEMa, вызовет фиксированный алгоритм (согласовать!!!), который вызовет этот алгоритм в свою очередь, и удалит этот файл. Такой механизм нужен, чтобы не менять текущее состояние файлов GMD и не иметь завязки на сетевое или удалённое расположение файлов алгоритмов.

Пример NOTIFY.ALG:

```
LOCAL curSender;
```

```
curSender=GetIDByMoniker('objref:...');
```

```
SetObjID GetIDByMoniker('objref:...');
```

```
CallAlg 600 curSender;
```

```
SetObjID GetIDByMoniker('objref:...');
```

```
CallAlg 600 curSender;
```

```
SetObjID GetIDByMoniker('objref:...');
```

```
CallAlg 600 curSender;
```

```
curSender=GetIDByMoniker('objref:...');  
  
SetObjID GetIDByMoniker('objref:...');  
CallAlg 600 curSender;  
SetObjID GetIDByMoniker('objref:...');  
CallAlg 600 curSender;  
End;
```